

Gilsiley Henrique Darú

**Uma Heurística para o Sequenciamento da Produção
Baseada na Teoria das Restrições**

**Dissertação apresentada como requisito parcial
à obtenção do grau de Mestre em Ciências,
Curso de Pós-Graduação em Métodos
Numéricos em Engenharia, Concentração em
Programação Matemática, Setor de Tecnologia,
Departamento de Construção Civil e Setor de
Ciências Exatas, Departamento de Matemática
da Universidade Federal do Paraná.**

Orientador: Prof. Dr. Celso Carnieri

CURITIBA

2005

Termo de Aprovação

Gilsiley Henrique Darú

Uma Heurística para o Sequenciamento da Produção Baseada na Teoria das Restrições

Dissertação aprovada como requisito parcial à obtenção do grau de Mestre em Ciências, Curso de Pós-Graduação em Métodos Numéricos em Engenharia, Concentração em Programação Matemática, Setor de Tecnologia, Departamento de Construção Civil e Setor de Ciências Exatas, Departamento de Matemática da Universidade Federal do Paraná, pela seguinte banca examinadora:

Orientador:

Prof. Dr. Celso Carnieri

Departamento de Matemática, UFPR

Prof. Dr. Reinaldo Morabito

Departamento de Engenharia de Produção, UFSCar

Prof. Dr. Sérgio Fernando Mayerle

Departamento de Engenharia de Produção, UFSC

Prof. Dr. Maria Teresinha Arns Steiner

Departamento de Matemática, UFPR

CURITIBA

2005

Dedicatória

A minha esposa Loreni e filho Charles, que privados em muitos momentos, souberam compreender e dar força para conclusão deste trabalho.

Aos meus pais, Darú e Dora, pela educação, exemplo e amparo.

Ao meu amigo Ayres, pelo incentivo e fomento à busca por desafios.

Epígrafe

A definição de conjuntos:

" Toda multiplicidade pode ser pensada como uma unidade, ou seja, toda coleção de elementos determinados pode ser, por uma lei, combinada em um todo ".

George Cantor (1845 - 1918).

Sumário

LISTA DE TABELAS.....	VIII
LISTA DE FIGURAS.....	IX
1 INTRODUÇÃO	1
1.1 OBJETIVOS DO TRABALHO	1
1.2 IMPORTANCIA DO TRABALHO	1
1.3 LIMITAÇÕES.....	2
1.4 ESTRUTURA DO TRABALHO	2
2 ADMINISTRAÇÃO DA PRODUÇÃO E OPERAÇÕES (APO)	3
2.1 ORIGENS	3
2.1.1 <i>Revolução Industrial (1700-1900)</i>	3
2.1.2 <i>Administração Científica (1850 – 1950)</i>	3
2.1.3 <i>Behaviorismo (1900 – 1950)</i>	4
2.1.4 <i>Pesquisa Operacional (1920 -)</i>	5
2.1.5 <i>Tecnologia da Informação</i>	5
2.2 CONCEITUAÇÃO	6
2.3 OBJETIVOS DOS ADMINISTRADORES DA PRODUÇÃO E OPERAÇÕES	7
2.4 HIERARQUIA DAS DECISÕES	8
2.4.1 <i>Decisões Estratégicas ou de Longo Prazo</i>	8
2.4.2 <i>Decisões Táticas ou de Médio Prazo</i>	8
2.4.3 <i>Decisões Operacionais e de Controle ou de Curto Prazo</i>	8
3 HIERARQUIA DO PLANEJAMENTO DA PRODUÇÃO	10
3.1 INTRODUÇÃO	10
3.2 DECISÕES ESTRATÉGICAS OU DE LONGO PRAZO.....	11
3.2.1 <i>Tipo de Produto</i>	12
3.2.2 <i>Foco da Produção</i>	12
3.2.3 <i>Política de Estoque dos Itens Acabados</i>	14
3.3 DECISÕES TÁTICAS OU DE MÉDIO PRAZO	15
3.3.1 <i>Planejamento Agregado</i>	15
3.3.2 <i>Plano Mestre de Produção</i>	16
3.4 DECISÕES OPERACIONAIS OU DE CURTO PRAZO	19
3.4.1 <i>Sistemas de Estoque de Reserva</i>	19
3.4.2 <i>Sistema Empurrar</i>	19
3.4.3 <i>Sistema Puxar</i>	20
3.4.4 <i>Manufatura Síncrona</i>	20
4 TEORIA DAS RESTRIÇÕES.....	21

4.1	INTRODUÇÃO A TOC	21
4.2	FILOSOFIA DE PENSAMENTO	22
4.2.1	<i>Componentes da Teoria das Restrições.....</i>	23
4.3	ETAPAS DE FOCALIZAÇÃO DA TOC	24
4.3.1	<i>Identificação da Restrição.....</i>	25
4.3.2	<i>Explorar a Restrição</i>	26
4.3.3	<i>Subordinar tudo à restrição</i>	26
4.3.4	<i>Elevar a restrição.....</i>	27
4.3.5	<i>Não deixar a inércia tomar conta, e voltar ao passo um</i>	27
4.4	INDICADORES DE DESEMPENHO DA TOC	27
4.4.1	<i>Ganho ou Throughput</i>	28
4.4.2	<i>Inventário</i>	29
4.4.3	<i>Despesa Operacional</i>	29
4.5	PROGRAMAÇÃO DA PRODUÇÃO	30
4.5.1	<i>Tambor</i>	30
4.5.2	<i>Pulmão</i>	31
4.5.3	<i>Corda.....</i>	32
5	SEQUENCIAMENTO	34
5.1	INTRODUÇÃO	34
5.2	DEFINIÇÃO.....	36
5.3	NOTAÇÃO	37
5.4	TERMINOLOGIA.....	37
5.4.1	<i>O Parâmetro α.....</i>	37
5.4.2	<i>O parâmetro β.....</i>	39
5.4.3	<i>O parâmetro γ.....</i>	41
5.5	REPRESENTAÇÃO	46
5.5.1	<i>Gráfico de Gantt.....</i>	46
5.5.2	<i>Grafos Disjuntivos.....</i>	47
5.6	COMPLEXIDADE	49
5.6.1	<i>Noções Gerais</i>	49
5.6.2	<i>Redução dos problemas de sequenciamento</i>	52
6	HEURÍSTICA PROPOSTA	53
6.1	INTRODUÇÃO	53
6.2	HIPÓTESES E CONSTATAÇÕES	54
6.3	LOCALIZAÇÃO DA FUNÇÃO SEQUENCIAMENTO	55
6.4	A HEURÍSTICA DE SEQUENCIAMENTO	62
6.4.1	<i>Cálculo de Prioridades</i>	62
6.4.2	<i>Designação de Ferramental.....</i>	65
6.4.3	<i>Designação de Centros de Trabalho.....</i>	74

6.4.4	<i>Sequenciamento de Tarefas</i>	83
6.4.5	<i>Agendamento</i>	92
7	RESULTADOS E CONCLUSÕES	94
7.1	CONCLUSÕES E CONTRIBUIÇÕES	94
7.1.1	<i>Teoria das Restrições</i>	95
7.1.2	<i>Programação Dinâmica</i>	96
7.1.3	<i>Heurística ATC modificada</i>	97
7.2	SUGESTÕES PARA TRABALHOS FUTUROS	103
7.2.1	<i>Sugestões de Aprimoramento</i>	104
7.2.2	<i>Sugestões de Comparação</i>	104
8	REFERÊNCIAS BIBLIOGRÁFICAS	106
	GLOSSÁRIO	108
	ANEXO A – CÓDIGO DE PROGRAMAÇÃO LINEAR INTEIRA UTILIZADO NO SOFTWARE LINGO	110
	ANEXO B – EXTERNALIZAÇÃO DE PARÂMETROS E TRANSFORMAÇÃO DA HEURÍSTICA ATC PARA OUTRAS HEURÍSTICAS	111
	ANEXO C – CÓDIGO FONTE DAS ROTINAS EM LINGUAGEM C, UTILIZADAS PARA SEQUENCIAMENTO	112

Lista de Tabelas

Tabela 2.1 Personagens principais da administração científica	4
Tabela 2.2 Alguns sistemas de produção.....	6
Tabela 2.3 Objetivos de desempenho	7
Tabela 3.1 Vinculando Estratégias de Posicionamento com Estratégias de Mercado	15
Tabela 3.2 Exemplo de um plano mestre de produção para a empresa fictícia de calçados	17
Tabela 4.1 Tabela de análise da capacidade da empresa	25
Tabela 4.2 Tabela de análise da capacidade da empresa após aumento de capacidade ..	27
Tabela 5.1 Resumo das funções objetivos derivadas	45
Tabela 6.1 Resumo das combinações dos parâmetros.....	65
Tabela 6.2 Tempos padrão das operações	78
Tabela 6.3 Tempo Processamento Operação x Centro de Trabalho.....	78
Tabela 6.4 Modelo tabular para solução do exemplo	79
Tabela 6.5 Disponibilidade dos os centros de trabalho	80
Tabela 6.6 Modelo tabular para solução do exemplo considerando disponibilidades	81
Tabela 6.7 Modelo tabular para solução do exemplo balanceando ferramentas	81
Tabela 6.8 Modelo tabular para solução do exemplo considerando ferramenta	82
Tabela 6.9 Dados para o exemplo 6.2	90
Tabela 6.10 Cálculo do ATC para as operações do exemplo 6.2, utilizando-se $k_1 = 0.5$ e $k_2 = 3$, $t = 0$	90
Tabela 6.11 Iterações para o sequenciamento das operações do exemplo 6.2	91
Tabela 6.12 Datas de início das operações	93
Tabela 7.1 Parâmetros e intervalos das distribuições das variáveis para simulação	100
Tabela 7.2 Resultados obtidos para $TF = 0.2$	101
Tabela 7.3 Resultados obtidos para $TF = 0.5$	102

Lista de Figuras

Figura 2.1 Esquema geral do papel da APO.....	6
Figura 3.1 Hierarquia do planejamento.....	11
Figura 3.2 Linha de produção com foco no produto - flowshop.....	13
Figura 3.3 Linha de produção com foco no processo - jobshop.....	13
Figura 3.4 Planejamento agregado	16
Figura 3.5 Distribuição da carteira dentro do horizonte de planejamento.....	18
Figura 4.1 Modelo para fabricação do item C	22
Figura 4.2 Componentes da Teoria das Restrições.....	24
Figura 4.3 Esquema de produção do produto C	25
Figura 4.4 Relacionamento entre indicadores globais e de produção	29
Figura 4.5 Identificação do tambor ou gargalo produtivo	31
Figura 4.6 Inserção dos pulmões de restrição, convergência e expedição	32
Figura 4.7 Ilustração das “cordas” conectando os “pulmões”	33
Figura 5.1 Localização da função sequenciamento dentro do planejamento da produção	35
Figura 5.2 Ilustração da função violação.....	42
Figura 5.3 Ilustração da função atraso.....	43
Figura 5.4 Ilustração da função antecipação	44
Figura 5.5 Ilustração da função <i>Just in Time</i>	44
Figura 5.6 Exemplo de um gráfico de Gantt	46
Figura 5.7 Representação do problema citado no texto, extraído de (Jain, 1988).....	48
Figura 5.8 Solução ótima para o problema da figura 5.7, extraído de (Jain, 1988)	49
Figura 5.9 Possível configuração para as classes \mathcal{P} , \mathcal{NP} e \mathcal{NP} Díficil.....	51
Figura 5.10 Árvore de redução dos problemas de sequenciamento	52
Figura 6.1 Fases de um sistema de planejamento fino, baseado na teoria das restrições.....	55
Figura 6.2 Interface de um sistema comercial de planejamento fino e suas fases	56
Figura 6.3 Informações disponíveis após os cálculos da fase de planejamento	57
Figura 6.4 Ordens geradas após a fase de explosão	58
Figura 6.5 Cálculos realizados durante a explosão para a rede de operações/lista de material gerada	58
Figura 6.6 Carga máquina para identificação do recurso com restrição de capacidade.....	59
Figura 6.7 Fases do sequenciamento	59

Figura 6.8 Interface de exploracao para aplicação de horas extras	60
Figura 6.9 Subordinação de operações não restritivas à restrição	61
Figura 6.10 Fases do Sequenciamento	62
Figura 6.11 Localização de uma operação para cálculo de sua prioridade	63
Figura 6.12 Exemplo do problema de balanceamento de ferramental	66
Figura 6.13 Representação das Variáveis.....	67
Figura 6.14 Exemplo de rótulo	68
Figura 6.15 Avaliação de alternativa.....	69
Figura 6.16 Exemplo de avaliação de alternativas	71
Figura 6.17 Aplicação do Algoritmo de designação de ferramental para a figura 6.12..	73
Figura 6.18 Trajetória final para o problema da figura 6.12	74
Figura 6.19 Designação após execução do algoritmo	74
Figura 6.20 Exemplo de definição de estágio e alternativa para designação de ferramental.....	76
Figura 6.21 Exemplo com a inclusão da restrição de centros de trabalho válido.....	77
Figura 6.22 Centros de trabalho e suas performances	78
Figura 6.23 Solução do exemplo, considerando performance dos centros de trabalho...	79
Figura 6.24 Solução considerando disponibilidade dos centros.....	80
Figura 6.25 Solução considerando-se ferramental	82
Figura 6.26 Relação de <i>ATC</i> com seus parâmetros	86
Figura 6.27 Relação <i>ATC</i> com seus parâmetros, porém considerando o atraso.....	87
Figura 6.28 Relação do <i>ATC</i> em função de <i>k</i> e número de operações	88
Figura 6.29 Esboço das folgas para as operações da tabela 6.9	90
Figura 6.30 Gantt das operações após aplicações da heurística <i>ATC</i> modificada.....	92
Figura 7.1 Amostra utilizando-se $TF = 0.2$	100
Figura 7.2 Amostra utilizando-se $TF = 0.5$	101

Uma Heurística para o Sequenciamento da Produção Baseada na Teoria das Restrições

Autor: Gilsiley Henrique Darú

Orientador: Prof. Dr. Celso Carnieri

Resumo

Em grande parte das empresas produtivas há problemas de fabricação, problemas estes derivados principalmente do mau uso dos recursos disponíveis. Isto é devido principalmente pelo empirismo inerente à tomada de decisão no chão de fábrica. Este fato justifica-se por se ter que trabalhar com inúmeras variáveis de decisão, tais como: data de entrega de um pedido, prioridade de cliente, recursos e ferramentais, disponibilidade de máquinas, feriados, calendários, ordens de produção firme, planejada entre outros. Assim o planejador deve tomar as decisões: “o que”, “quando”, “onde” e utilizando quais recursos” para produzir. Sendo assim este trabalho propõe uma heurística modificada, adaptada de (Rachamadugu, 82), que se utiliza da teoria das restrições para simplificar o número de variáveis, objetivando a minimização do atraso ponderado, em problemas de sequenciamento da produção que leva em conta prioridade, calendário, performances, ferramentais, centros paralelos, datas mínimas e datas mais tarde para realizar o planejamento.

Palavras-chave: Heurística, Regra de Despacho, ATC, Atraso Ponderado, Sequenciamento, Teoria das Restrições, Planejamento Fino da Produção.

A Heuristic for Production Scheduling Based in Theory of Constraints

Author: Gilsiley Henrique Darú

Adviser: Prof. Dr. Celso Carnieri

Abstract

Many companies have problems with you're their production process. These problems come mainly from the poor resources available, specially due to decisions in shop floor. There are many variables for taking decisions, for example: due date for a order, its priority, the clients priority, free resources, tools, time disponible in machines, holidays, calendars, order status, and so on. Thus, the planner must make decisions of what, when, and where, using what to product. So this work will develop a new modified heuristic based in theory of constraints to minimize the weighted tardiness in scheduling problems, seeing calendar, machine's performance, tools, parallel machines, release dates and due dates.

Key Words: Heuristic, Dispatch Rule, Apparent Tardiness Cost, Weighted Tardiness, Scheduling, Theory of Constraints, Fine Production Planning.

1 Introdução

Este trabalho nasceu de uma necessidade de implementação em um sistema comercial de técnicas de sequenciamento. Este sistema devia se utilizar da teoria das restrições e em uma de suas fases sequenciar operações.

Este sequenciamento deve respeitar prioridades, datas de entrega e outras restrições do chão de fábrica como seqüências tecnológicas e utilização de ferramentas produtivas.

O trabalho, portanto, visa implementar uma técnica de sequenciamento de operações, embutido em um sistema de planejamento da produção, que respeite restrições e execute em um tempo computacional que permita em um breve espaço de tempo realizar simulações de programação.

1.1 *Objetivos do Trabalho*

Os objetivos do trabalho são:

a) Desenvolver uma heurística para o sequenciamento da produção adaptável ao chão de fábrica. Esta heurística designa operações a centros de trabalho objetivando equilibrar suas utilizações. Designa ferramentas conduzindo ao equilíbrio de suas cargas de utilização. Sequencia operações levando em conta as prioridades das operações, suas datas de entrega, suas datas mais cedo visando minimizar os atrasos das operações prioritárias.

b) Estudar os efeitos da aplicação da teoria das restrições. Identificar os aspectos positivos e negativos ao se sequenciar apenas as restrições produtivas. Isto é, dado um conjunto de recursos produtivos, identificar o recurso com a menor capacidade ou a capacidade mais utilizada, e sequenciá-lo, forçando os demais centros produtivos a obedecer este sequenciamento.

1.2 *Importância do Trabalho*

O principal aspecto relevante do trabalho é unir a teoria com a prática. Sua implementação em um sistema comercial, baseando-se em modelos teóricos e incorporando uma teoria comercial, a teoria das restrições.

Outra contribuição importante é referente ao pioneirismo de trabalhos nesta área, no Brasil, que trata de sequenciamento e teoria das restrições.

1.3 Limitações

Devido à complexidade do problema de sequenciamento, e justificado pelo uso da teoria das restrições, o trabalho limita-se a sequenciar apenas as operações restritivas, cabendo a teoria das restrições a imposição dos inícios das operações não restritivas.

Outra limitação faz jus ao não tratamento do setup de máquinas, o que é sugerido para trabalhos futuros.

1.4 Estrutura do Trabalho

Para uma melhor compreensão e escolha de qual técnica a ser utilizada, bem como contextualizar e entender a localização do sequenciamento dentro da produção decidiu-se estruturá-lo de forma que os primeiros três capítulos tratam de aspectos teóricos. Realiza-se, no capítulo 2, uma fundamentação teórica e faz-se uma revisão dos aspectos históricos da administração da produção, As fases do planejamento da produção são comentadas no capítulo 3, de modo a situar o sequenciamento dentro da produção. O capítulo 4 trata de expor a teoria das restrições e sua aplicação dentro do cenário produtivo.

A partir do quinto capítulo, explana-se sobre os aspectos referentes ao sequenciamento, realizando uma explanação teórica sobre o mesmo, bem como definindo, classificando e situando-o dentro do contexto produtivo.

O sexto capítulo explana sobre a metodologia proposta, descrevendo passo a passo todas as questões referente a solução do problema e como esta foi implementada.

O capítulo 7 aborda os resultados obtidos, as considerações e comentários relativos à solução, bem como sugere trabalhos a futuros.

2 Administração da Produção e Operações (APO)

2.1 Origens

As civilizações antigas como os romanos, gregos e egípcios, já confeccionavam produtos para escambo e construíam grandes obras como o Coliseu, as Pirâmides e o Parthenon. Estes povos precisavam se organizar para atingir suas metas, a construção de suas obras. Devido a isto, pode-se considerá-los como precursores da *APO*.

Atualmente a *APO* é considerada uma disciplina aplicada. Mas para compreender como se encontra hoje, necessita-se analisar alguns períodos importantes os quais marcaram época e alicerçaram novas bases a esta disciplina. Estas fases podem ser divididas em cinco, tomando como base (Gaither et al, 1999) e (Davis et al, 2003) e adaptando a realidade brasileira.

2.1.1 Revolução Industrial (1700-1900)

Surgem as primeiras fábricas, devido ao desenvolvimento do motor e a consequente substituição da força humana pela força mecanizada. Adam Smith publica o livro *A riqueza das nações* e difunde a idéia da divisão do trabalho e especialização da mão-de-obra. Ainda neste período Ely Whitney desenvolve o conceito de peças intercambiáveis. Com todo este desenvolvimento até então inédito, começa-se a migrar do artesanato, ou produção caseira, para a ciência da produção.

2.1.2 Administração Científica (1850 – 1950)

O principal representante deste período foi Frederick Winslow Taylor. Considerado o pai da administração científica, ele estudou cientificamente os problemas fabris de sua época e popularizou a noção de **eficiência** — obter o resultado desejado com o menor desperdício de tempo, esforço e materiais. Outros autores se destacam neste período e a tabela 2.1 descreve estes personagens e suas contribuições. Para finalizar este período do "boom" da APO, outro grande protagonista foi Henry Ford, que concebeu as linhas de montagem, tornando as fábricas como “rios e seus afluentes” e desenvolveu o conceito **produção em massa**.

Tabela 2.1 Personagens principais da administração científica

Contribuinte	Período	Contribuições
Frederick Winslow Taylor	1856-1915	Princípios de administração científica, princípio da exceção, estudo do tempo, análise de métodos, padrões, planejamento e controle.
Frank B. Gilbreth	1868-1934	Estudo dos movimentos, métodos, contratos de construção, consultoria.
Lillian M. Gilbreth	1878-1973	Estudos da fadiga, ergonomia, seleção e treinamento de empregados.
Henry L. Gantt	1861-1919	Gráficos de Gantt, sistemas de pagamento por incentivo, abordagem humanística ao trabalho, treinamento.
Carl G. Barth	1860-1939	Análise matemática, régua de cálculo, estudos de suprimento e velocidade, consultoria para a indústria automobilística.
Harrington Emerson	1885-1931	Princípios da eficiência, métodos de controle.
Morris L. Cooke	1872-1960	Aplicação da administração científica à educação e ao governo.

Fonte: Adaptado pelo autor de (Gaither et al, 1999)

Ainda neste período, surgiu o termo “engenheiro de produção”. Segundo a *ABEPRO* (Associação Brasileira de Engenharia de Produção) e *IIIE* (*International Institute of Industrial Engineering*) (Abepro, 2005) “*competir ao engenheiro de produção o projeto, a implantação, a operação, a melhoria e a manutenção de sistemas produtivos integrados de bens e serviços, envolvendo homens, materiais, tecnologia, informação e energia. Competir ainda especificar, prever e avaliar os resultados obtidos destes sistemas para a sociedade e o meio ambiente, recorrendo a conhecimentos especializados da matemática, física, ciências humanas e sociais, conjuntamente com os princípios e métodos de análise e projeto da engenharia*”.

2.1.3 Behaviorismo (1900 – 1950)

Este período caracterizou-se pela identificação de quais fatores humanos poderiam influenciar na produção. O precursor foi Elton Mayo que junto com sua equipe da *Harvard Business School* conduziu os estudos de *Hawthorne*. Estes estudos objetivaram identificar níveis ótimos de iluminação. Como os resultados foram confusos, a conclusão foi que fatores comportamentais (*behavior*) tinham influência nos trabalhadores. Por isto este período ficou conhecido como behaviorismo. Ainda hoje se busca entender melhor a “máquina” humana para fazê-la “produzir” mais.

2.1.4 Pesquisa Operacional (1920 -)

Com o aumento da complexidade na tomada de decisão e com necessidades reais de desenvolver uma ferramenta para auxiliar neste processo, surge a Pesquisa Operacional. Uma ciência de caráter interdisciplinar que se utiliza de métodos quantitativos para identificar a melhor resposta. Visa substituir a tomada de decisão **intuitiva** por um método que encontra a resposta ótima ou quase ótima. Tem como principais características:

- ☑ Visão Holística
- ☑ Interdisciplinaridade
- ☑ Utilização intensiva de computadores
- ☑ Constrói modelos para realizar experiências
- ☑ Principal objetivo é melhorar o processo de decisão

2.1.5 Tecnologia da Informação

Uma nova era se caracteriza com a utilização dos computadores como auxiliares na *APO*. Com o advento de técnicas como *MRP* I e II e *CRP* que requerem uma enorme quantidade de cálculos, passou a ser obrigatório o uso intensivo do computador. Isto fez emergir os sistemas *ERP*, com o objetivo de integrar horizontalmente uma empresa. As áreas de recursos humanos, finanças, logística, relacionamento com clientes, marketing, manufatura, entre outras, devido à facilidade de armazenamento de dados, passam a ter informação e tomar melhores decisões. Há ainda os sistemas de planejamento fino da produção e sistemas especialistas. Passos como a automação industrial interligada ao *ERP*, pesquisa operacional embutida já nas ferramentas de tomada de decisão, informação *on-line* via internet, palm-tops, rádio-frequência e celulares são as promessas do avanço para os anos seguintes¹.

Com um perpassar sobre os fatos históricos que conduziram ao surgimento da *APO*, bem como caracterizaram suas fronteiras e deram-lhe forma, faz-se necessário agora definir de maneira geral o que é a administração da produção.

¹ Informações levantadas junto a uma empresa fornecedora de ERP do Brasil.

2.2 Conceituação

Administrar a produção e as operações, conforme (Davis et al, 2003) é “*administrar o processo de conversão que transforma insumos tais como matéria-prima ou esforço em saídas sob a forma de bens acabados ou serviços*”.

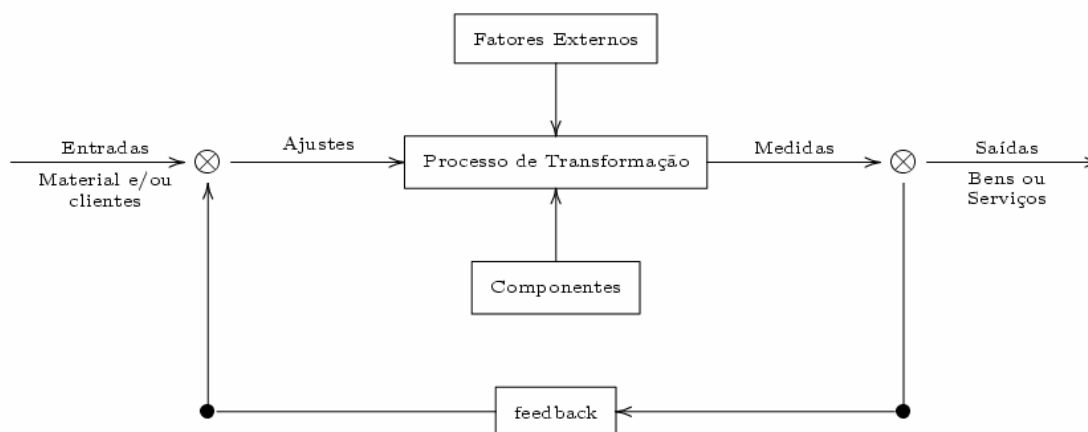


Figura 2.1 Esquema geral do papel da APO

Fonte: Adaptado pelo autor de (Slack e Nigel, 1997)

Muitos autores concordam com esta definição tais como as referências (Gaither, 1999) e (Slack e Nigel, 1997). A figura 2.1 exhibe um esquema geral da função produção. Este esquema delineia três componentes principais e dois de ajuste: as entradas, o processo de transformação e as saídas. Como complementares tem-se o processo de medição das saídas e os ajustes das entradas. As empresas de manufatura têm em sua essência este arquétipo, porém grande parte das empresas de serviço não. Isto não significa que elas não possam se enquadrar neste modelo, pelo contrário, é apenas uma questão de determinação e identificação correta das entradas, processamentos e saídas. A tabela 2.2 exhibe alguns exemplos.

Tabela 2.2 Alguns sistemas de produção

Sistema de Produção	Insumos Primários	Subsistema de Transformação	Saídas (Produtos)
Porto de containers	Navios e cargas, funcionários e equipamento de movimentação de containers.	Movimento de cargas do navio para o cais e vice-versa	Navios carregados ou descarregados
Colégio ou universidade	Estudantes, livros, suprimentos, pessoal, prédios, serviços públicos.	Transmitir e gerar conhecimento	Pessoas instruídas, novas tecnologias
Funilaria	Carros batidos, danificados, com problemas na pintura.	Recupera o que foi danificado deixando igual ao original	Carros recuperados

Uma vez conceituada a APO, faz-se necessário identificar seu papel dentro das empresas e alinhá-lo com os objetivos corporativos.

2.3 Objetivos dos Administradores da Produção e Operações

Em (Gaither et al, 1999) afirma-se que “*para obter sucesso à competição global, as empresas devem ter um compromisso com a receptividade do cliente e com a melhoria contínua rumo à meta de desenvolver rapidamente produtos inovadores que tenham a melhor combinação de excepcional qualidade, entrega rápida e no tempo certo, com preços e custos baixos*”. Observando esta afirmação percebe-se a diversidade de papéis e objetivos que as empresas buscam, bem como os desafios que estas têm pela frente.

Baseado nisto, pode-se identificar alguns objetivos gerais que norteiam os administradores da produção e operações. Extraído de (Gaither, et al, 1999), eles são:

- ☑ Qualidade;
- ☑ Rapidez;
- ☑ Confiabilidade;
- ☑ Flexibilidade;
- ☑ Custo;

De acordo com as estratégias da empresa, esta pode obter vantagens competitivas em um ou mais quesitos. Quanto mais vantagem, mais inovação, tecnologia, investimento e criatividades devem ser investidos. Estas vantagens extraídas de (Slack e Nigel, 1997) podem ser resumidas na tabela 2.3.

Percebe-se que estes objetivos são muito amplos em termos empresariais. Para atingi-los, o administrador deve particionar o problema em partes menores. Esta divisão facilita o trabalho, pois permite concentrar-se em uma parte de cada vez. No caso da APO, esta divisão é hierarquizada e cada uma é baseada em decisões dependentes do horizonte de tempo a ser considerado. O tópico a seguir trata desta hierarquia.

Tabela 2.3 Objetivos de desempenho

Fazer certo as coisas	►	Vantagem em qualidade
Fazer as coisas com rapidez	►	Vantagem em rapidez
Fazer as coisas em tempo	►	Vantagem em confiabilidade
Mudar o que faz	►	Vantagem em flexibilidade
Fazer as coisas mais baratas	►	Vantagem em custo

2.4 Hierarquia das Decisões

Considerando que a *APO* é uma função de produção, as decisões gerenciais nesta função podem ser divididas em três grandes áreas definidas a seguir.

2.4.1 Decisões Estratégicas ou de Longo Prazo

Estas decisões dizem respeito ao futuro de uma empresa, e servem como restrições aos outros planejamentos. As principais perguntas que são respondidas durante esta fase são:

- ☒ Quais produtos serão fabricados?
- ☒ Onde se construirá as fábricas e outras instalações?
- ☒ Qual a capacidade que será necessária?
- ☒ Quando será necessário aumentar a capacidade?

2.4.2 Decisões Táticas ou de Médio Prazo

Após definida a estratégia, as vendas começam a ocorrer e o sistema começa a funcionar. Isto desloca as preocupações para a fábrica de forma a responder as seguintes perguntas:

- ☒ Quantos funcionários serão necessários?
- ☒ Quando serão necessários estes profissionais?
- ☒ Usam-se horas extras ou criam-se novos turnos?
- ☒ Quando os materiais devem ser entregues?
- ☒ Quais os níveis de estoque necessários?

2.4.3 Decisões Operacionais e de Controle ou de Curto Prazo

É neste ponto que se inicia o interesse pelo sequenciamento das operações e, depois de respondidas as perguntas estratégicas e táticas, chega-se ao planejamento de curto prazo que indaga o seguinte:

- ☒ Quais tarefas devem ser executadas hoje ou nesta semana?
- ☒ Onde estas tarefas devem ser processadas?
- ☒ Quais tarefas são prioritárias?
- ☒ Quando se deve iniciar uma determinada tarefa?

Cada uma destas decisões conduz a um conjunto de restrições que vão determinar o modelo de sequenciamento que será utilizado. Devido a conceitos

importantes que se contextualizam dentro destes tópicos o capítulo seguinte trata de conceituar em mais detalhes o vocabulário inerente em cada uma destas fases.

3 Hierarquia do Planejamento da Produção

3.1 Introdução

Nos dias de hoje, as empresas podem afirmar que sua concorrência não é sua vizinha, mas sim pertencente ao que se chama de “aldeia global”. Esta internacionalização resulta em forte pressão externa sobre as corporações. Para reagir a esta pressão, estratégias bem definidas são diferenciais. Porém, não são somente estes fatores que influenciam. Segundo (Gaither et al, 1999) pode-se acrescentar o aumento da qualidade com diminuição dos custos, tecnologia de produção, crescimento do setor de serviços, escassez de recursos e responsabilidade social.

Portanto, refletir e tomar decisões melhores é uma questão chave para determinar a permanência ou desaparecimento de uma empresa no mercado. A figura 3.1 exibe uma divisão da tomada de decisão, dependente do horizonte de planejamento. Nos tópicos seguintes são abordadas quais decisões são estas e em que nível ou momento devem ser tomados.

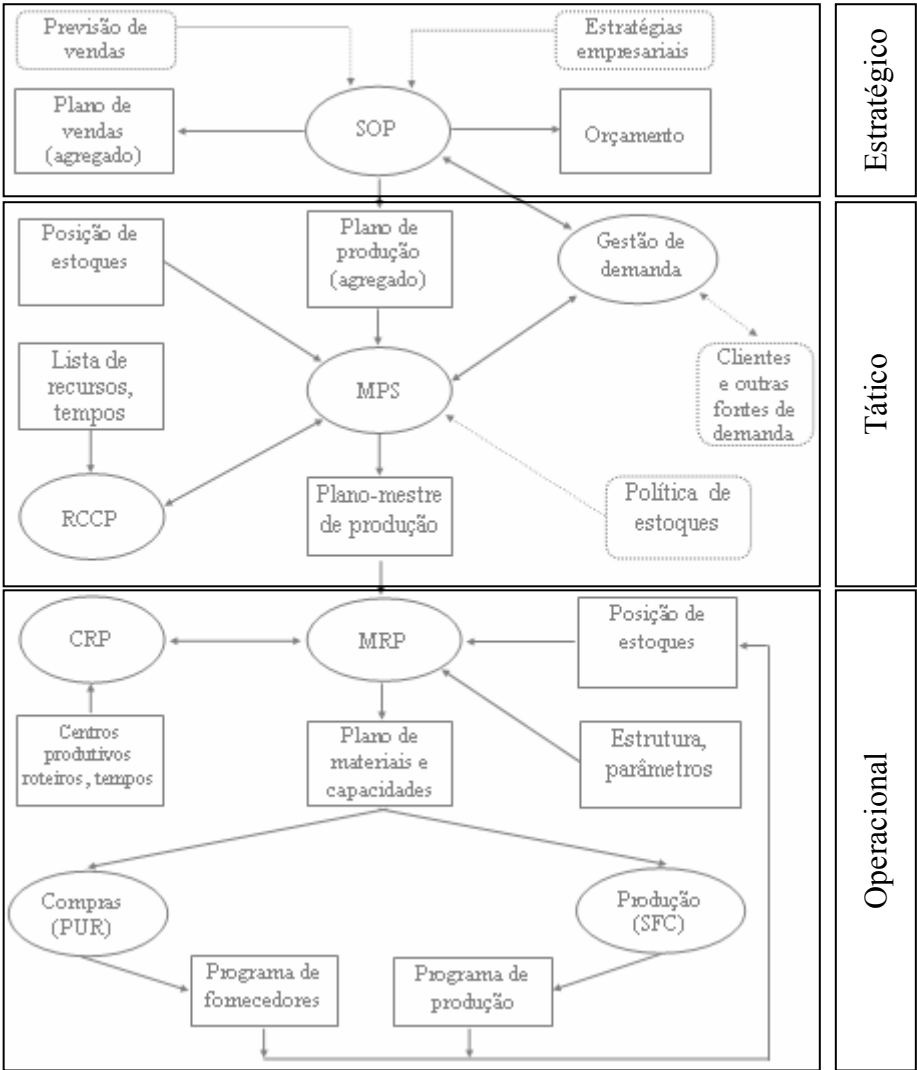


Figura 3.1 Hierarquia do planeamento

3.2 *Decisões Estratégicas ou de Longo Prazo*

Estas decisões traçam a estratégia de operações de uma empresa em longo prazo. Esta estratégia leva em conta questões como: quais novos produtos lançar, quando entrar em produção, quais e quando novas instalações serão necessárias, quais e quando novas tecnologias de produção devem ser desenvolvidas e quais serão os leiautes² de produção.

A seguir é detalhado o posicionamento do sistema de produção. As decisões tomadas nesta etapa impactam diretamente em como a empresa vai reagir ao mercado. Estas decisões dizem respeito ao tipo de produto, o volume de produção, o foco da produção e política de estoque.

² Do inglês layout, termo já adaptado para o português.

3.2.1 Tipo de Produto

Questões como variabilidade, diversidade e volumes de produção são definidos de acordo com o tipo de produto a se produzir. Podem ser padrões ou personalizados.

Produto padrão

Neste tipo tem-se pouca variabilidade e diversidade. Por consequência a produção será de altos volumes e dedicada. Estas características podem determinar o foco da produção voltado ao produto.

Produto personalizado

Ao contrário do anterior, a variabilidade e diversidade são grandes. Isto implica em provável produção com baixos volumes. Determina o foco da produção voltado ao processo, devido exatamente a variabilidade de itens em produção.

3.2.2 Foco da Produção

Esta questão determina, entre outras, o leiaute produtivo, e algumas características produtivas. Classifica-se, basicamente, em produto e processo.

Foco no Produto

Uma vez definido que o foco será no produto, a produção será voltada a prover altos volumes (em sua maioria). Para isto, são criadas as linhas de produção ou *flowshop*. Um exemplo típico são as montadoras de veículos. A figura 3.2 exibe um exemplo. O termo *flow* (fluxo) é derivado da semelhança entre este modelo e um rio que o fluxo vai de montante a jusante sem ser interrompido. Durante o percurso seus afluentes "despejam" mais fluido, ou seja, material, até o objetivo ser atingido (o item final produzido).

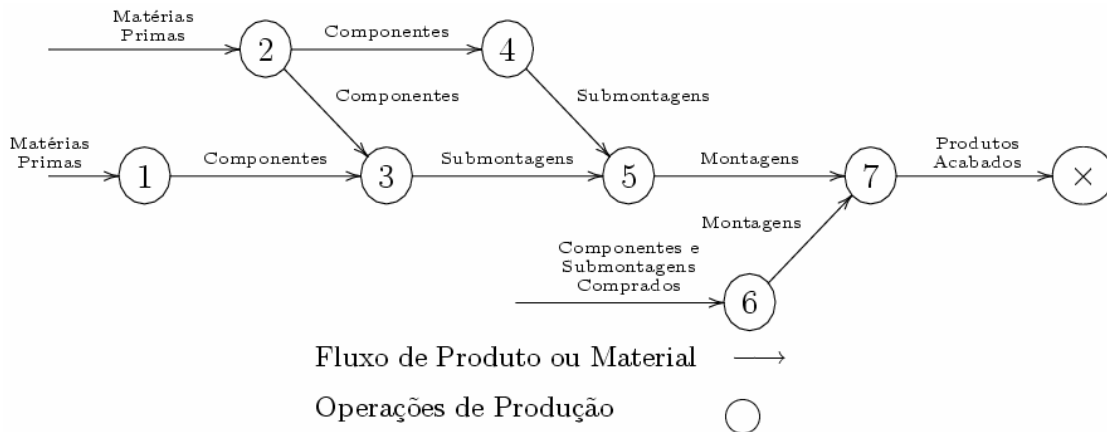


Figura 3.2 Linha de produção com foco no produto - flowshop

Foco no Processo

Neste caso, a produção terá uma diversidade de itens a serem produzidos. Isto leva não mais a definir uma linha de produção específica, mas sim definir quais as similaridades entre os itens e quais máquinas serão necessárias. Este grupo de máquinas, ou "oficina" é conhecido como *jobshop*. O foco não é mais no produto e sim no processo. Para se ter uma idéia de como é uma empresa com este foco, a figura 3.3 exibe o leiaute que adota o foco no processo. Neste modelo os produtos deslocam-se entre grupos de máquinas. Um item pode ir e voltar, percorrendo a fábrica, dependendo de quais operações são necessárias para completá-lo. Por exemplo, a operação rebarbar está no processo tanto do produto a quanto do produto b. Idem para os processos de embalagem, montagem, fabricação e cisalhamento.

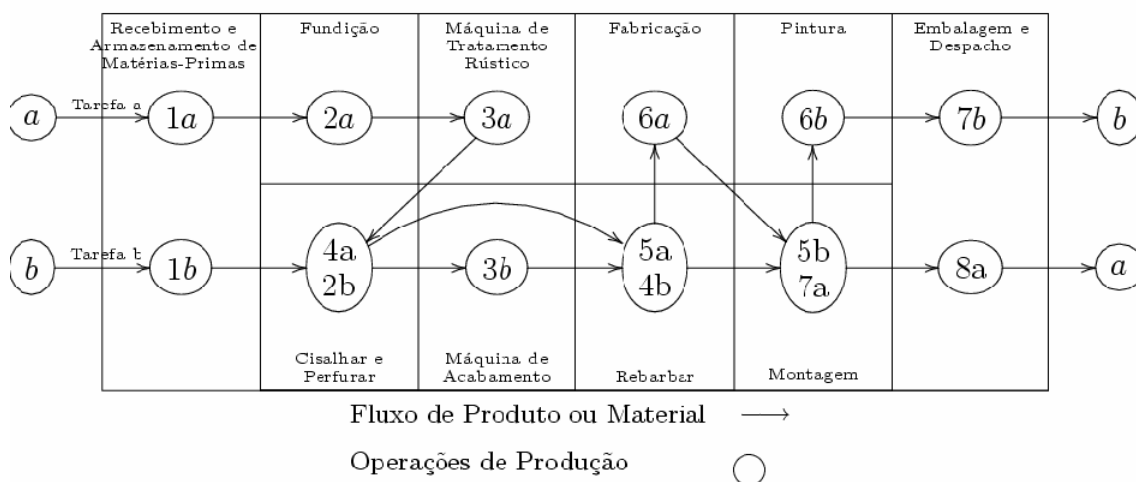


Figura 3.3 Linha de produção com foco no processo - jobshop

3.2.3 Política de Estoque dos Itens Acabados

Conforme visto anteriormente, outra decisão inerente ao posicionamento da produção é sua política de estoque com relação aos seus itens acabados. Esta pode ser basicamente de dois tipos: produzir para estoque *Make To Stock - MTS* ou produzir sob encomenda *Make To Order - MTO*.

Produzir para estoque - MTS

Fabricar para preencher o estoque é uma prática comum. Sempre que de antemão pode-se prever itens que serão vendidos, estes podem aproveitar momentos de entre-safra para serem produzidos, utilizando melhor os recursos e os carregando de maneira mais equilibrada. Assim, quando um item for solicitado por um cliente sua entrega se dá quase que imediatamente, pois o tempo considerado é apenas o de entrega ao cliente. A desvantagem desta política, para grande parte dos tipos de produto, é o alto custo de armazenagem e a dificuldade de acerto do que será vendido.

Produzir sob encomenda – MTO

Para evitar os custos de armazenagem, é possível que um item apenas seja produzido quando da entrada de um pedido ou uma encomenda. Isto gerará uma ordem de produção que envia à produção um sinal para começar a manufatura do item vendido. Os efeitos são contrários a política anterior. Aqui o tempo para a entrega ao cliente é dado pelo tempo de fabricação e até mesmo de compra de matéria-prima.

Estes são dois extremos e percebe-se que as indústrias brasileiras tendem a trabalhar com um modelo híbrido, onde os itens mais vendidos possuem a política *MTS* enquanto itens de baixo volume de vendas se utilizam do modelo *MTO*.

Para encerrar a discussão sobre estas estratégias de posicionamento de mercado das empresas, faz-se um resumo apresentado na tabela 3.1. Esta mostra algumas relações entre as estratégias de posicionamento com as estratégias de mercado.

Tabela 3.1 Vinculando Estratégias de Posicionamento com Estratégias de Mercado

Algumas estratégias comuns de posicionamento	Produtos personalizados		Produtos padronizados	
	Volume baixo	Volume elevado	Volume baixo	Volume elevado
Focalizado no produto para estoque				Competição baseada amplamente no custo de produção, na entrega rápida de produtos e na qualidade. Exemplo: televisores
Focalizado no produto sob encomenda			Competição baseada amplamente no custo de produção, em manter promessas de entrega e na qualidade. Exemplo: ônibus escolares	
Focalizado no processo para estoque		Competição baseada amplamente na flexibilidade, na qualidade e na rápida entrega dos produtos Exemplo: instrumentos médicos		
Focalizado no processo sob encomenda	Competição baseada amplamente na flexibilidade e em manter as promessas de entrega. Exemplo: supercomputadores			

3.3 Decisões Táticas ou de Médio Prazo

3.3.1 Planejamento Agregado

No planejamento estratégico decidiu-se pelo comportamento principal da empresa. É momento de, baseado na política estratégica, preparar-se para atender o mercado. Desta forma, para efeitos de compra de matéria-prima, negociação com fornecedores, tratamento de sazonalidades, entre outros, a empresa deve definir a contratação ou demissão de funcionários (para controle da capacidade). O estoque que deverá ser mantido (exemplo de períodos de entre-safra, Páscoa e Natal), modificações nas instalações e contratos com fornecedores são outras decisões a serem tomadas nesta fase. Este planejamento comumente recebe o nome de **planejamento agregado**.

Na figura 3.4 tem-se um exemplo de um planejamento agregado. Uma empresa fictícia que produz calçado está fazendo uma análise de demanda agregada. Após estudos levantaram-se as necessidades de solado (parte inferior do calçado) para os três produtos principais da empresa. Os gerentes sabendo que possuem uma capacidade máxima de 50000 unidades, e prevendo que no segundo trimestre ocorrerá uma sobrecarga, podem antecipadamente decidir o que farão. Citando-se alternativas como antecipar a produção no primeiro trimestre ou pagar horas extras.

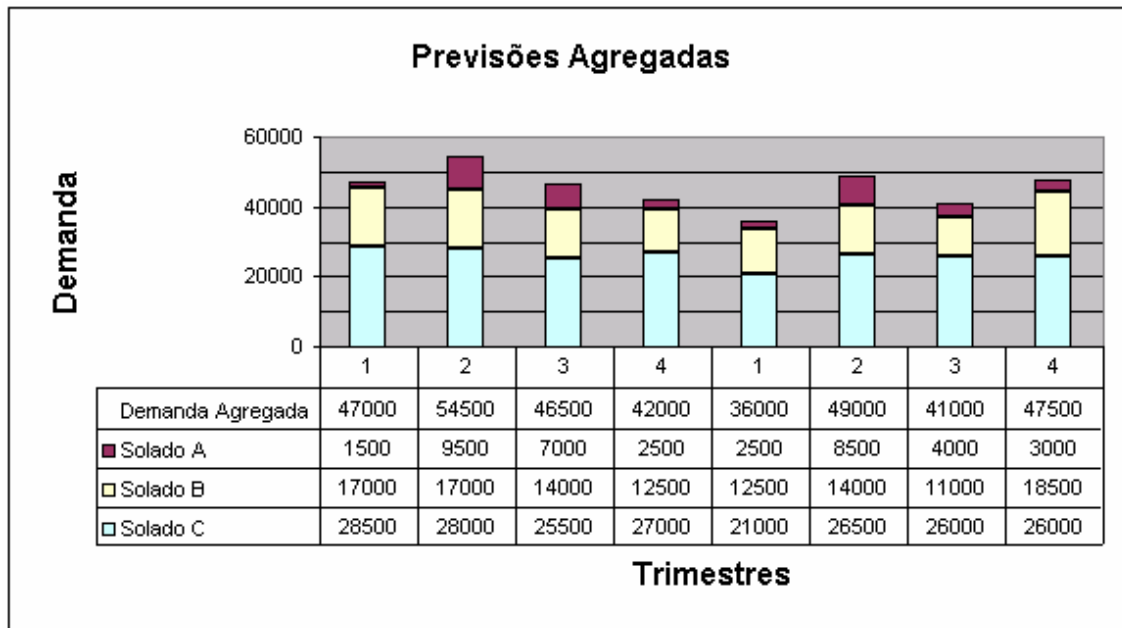


Figura 3.4 Planejamento agregado

Mas o planejamento tático pode ainda possuir um horizonte mais curto, como por exemplo, semanas. A este se dá o nome de **plano mestre de produção**.

3.3.2 Plano Mestre de Produção

Antes de se definir detalhes do dia-a-dia da empresa, deve-se primeiro desenvolver o que se chama de plano mestre de produção (*MPS - Master Program Schedule*). Este plano define a quantidade de cada item **final** a ser concluída em cada semana dentro do horizonte de planejamento³.

A tabela 3.2 exhibe um exemplo de um programa mestre de produção. Novamente tomando como exemplo a empresa fictícia de calçados da figura 3.4, tem-se não mais uma análise trimestral, mas sim uma previsão que se estende da semana 1 a

³ Ver glossário.

semana 8. A primeira semana corresponde a semana atual ou a semana seguinte, dependendo da preferência do planejador. Para um exemplo de cálculo, o solado do tipo A possui um estoque inicial [1] de 150 unidades. Estima-se uma demanda de 100 [2]. O saldo é obtido a partir do estoque inicial menos a demanda estimada [1]-[2]. O valor resultante 50 é confrontado com o estoque de segurança. Para o caso deste produto, este valor é dado a esquerda pelo número 100. Se o valor calculado é menor que este estoque, então se dispara uma produção conforme o tamanho de lote fixo, que para este produto é 200. A produção necessária é então 200 unidades, gerando um estoque final de 250 unidades. O processo seguido é o mesmo para as semanas seguintes e para os demais produtos.

Tabela 3.2 Exemplo de um plano mestre de produção para a empresa fictícia de calçados

Item Final		Semanas							
		1	2	3	4	5	6	7	8
Solado A	Estoque Inicial	150	250	150	250	250	250	250	250
	Demanda Estimada	100	100	100	200	200	200	0	100
Est. Seg	Saldo	50	150	50	50	50	50	250	150
100	Produção Necessária	200	0	200	200	200	200	0	0
200	Estoque Final	250	150	250	250	250	250	250	150
Solado B	Estoque Inicial	2000	2500	2000	1000	1500	2500	0	1500
	Demanda Estimada	1500	500	3000	1500	1000	2500	500	1000
Est. Seg	Saldo	500	2000	-1000	-500	500	0	-500	500
1000	Produção Necessária	2000	0	2000	2000	2000	0	2000	2000
2000	Estoque Final	2500	2000	1000	1500	2500	0	1500	2500
Solado C	Estoque Inicial	2000	2000	2000	1000	500	1500	1000	2000
	Demanda Estimada	2500	2500	3500	3000	1500	3000	1500	1000
Est. Seg	Saldo	-500	-500	-1500	-2000	-1000	-1500	-500	1000
2000	Produção Necessária	2500	2500	2500	2500	2500	2500	2500	2500
2500	Estoque Final	2000	2000	1000	500	1500	1000	2000	3500

O *MPS* deve ser revisado a cada período definido, de forma a torná-lo mais realista. Os primeiros períodos são dominados por pedidos concretos, porém para períodos mais afastados, o domínio é das previsões realizadas.

Como as revisões são constantes, se houver muita alteração neste plano, o caos pode dominar a produção. Para evitar este tipo de problema define-se períodos de congelamento ou *time fence*. Em (Gaither et al, 1999) divide-se os períodos de congelamento em **congelado**, as primeiras semanas do *MPS* compreendidas principalmente por pedidos já na iminência de serem entregues; **firme**, as próximas semanas após o período congelado, também compreendido por pedidos de clientes e não deve sofrer muitas alterações; **cheia**, logo após o período firme, compreende pedidos e previsões e ocupam toda capacidade produtiva (daí o porquê de cheia) e por fim **aberta**,

que são as últimas semanas do *MPS*, compreendida principalmente por previsões e ainda possuem capacidade produtiva a ser alocada. A figura 3.5 exibe os períodos do *time fence* bem como sua composição em relação a pedidos e previsões.

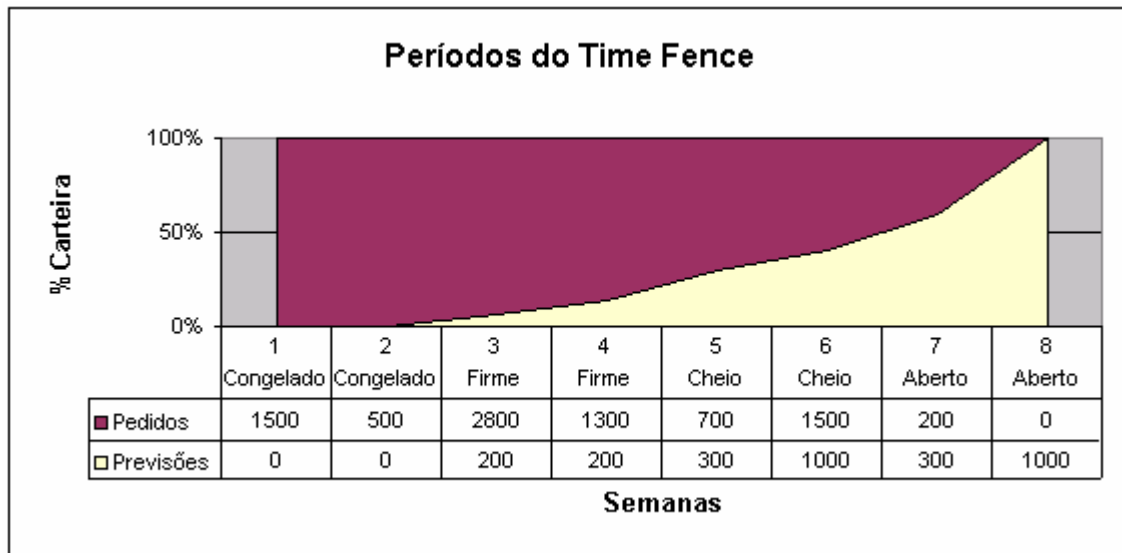


Figura 3.5 Distribuição da carteira dentro do horizonte de planejamento

Efetuada o *MPS*, é possível realizar uma análise sobre a carga dos centros de trabalho. Essa é uma verificação preliminar e serve para analisar se haverá sobrecarga ou subcarga em determinado período. A esta análise comumente chama-se planejamento da capacidade de médio prazo (**RCCP** - *Rought-Cut Capacity Planning*) que pode indicar alterações no *MPS*, ao se nivelar a capacidade.

Todos os itens analisados durante o planejamento do *MPS* são considerados do tipo **demanda independente** e por conseqüência, tem uma grande variabilidade. As previsões são baseadas em históricos, tendências e sazonalidades. Usam a estatística como suporte para estimar a variância, determinar erros máximos aceitáveis entre outros. Isto explica o porquê da necessidade constante de revisão e estimativas a cada período, para determinar a quantidade a produzir de cada item deste tipo (demanda independente).

Uma vez que se tenha em mãos os pedidos que devem ser atendidos e determinados os itens a serem produzidos, devem-se então passar as fases seguintes, que determinam todos os itens conhecidos como **demanda dependente**. Estes são os itens que fazem parte da estrutura de produto. Para realizar tal estimativa a técnica mais difundida é conhecida como planejamento das necessidades de materiais (**MRP** -

Materials Resource Planning). Outras técnicas, porém, têm ganhado relevância, como o *Just in Time - JIT* e a Teoria das Restrições (*Theory of Constraints – TOC*). Cada uma destas técnicas possui um enfoque dentro da manufatura.

3.4 Decisões Operacionais ou de Curto Prazo

Depois de concluído o *MPS*, sabe-se “o que” se deve produzir e para quando. Deste ponto em diante a organização foca-se em como realizar este plano. Para isto deve determinar “o quanto” e “para quando” deve comprar os materiais. Quando deve produzir e montar as peças intermediárias e em que seqüência deve ser feito. Estas respostas dependem do tipo de sistema de planejamento e controle que a empresa adota. Existem vários modelos, em (Gaither et al, 1999) são abordados quatro sistemas.

3.4.1 Sistemas de Estoque de Reserva

Esta é uma das abordagens mais simples, e necessita de poucas informações para ser utilizada. Basicamente, consiste em produzir para estoque, sendo este a reserva. Quando os clientes efetuam a compra, este estoque é diminuído e por conseqüência são geradas ordens à produção para repor o que foi comprado. Esta abordagem é fortemente dependente da política de estoque dos itens de demanda independente. Tem como características principais, o estoque excessivo de produtos acabados e materiais em processo e a inflexibilidade de resposta a pedidos que não estejam em estoque (pois primeiro deve-se repor o "buraco" no estoque, que já está em produção, para somente depois entrar em produção o pedido do cliente).

3.4.2 Sistema Empurrar

O *MRP* é a principal técnica utilizada neste modelo. Ele determina a partir do *MPS*, quando cada material deve estar disponível e quando deve ser comprado. Uma vez determinadas as datas, o sistema se inverte. Após a chegada de matéria-prima, o sistema começa o processamento e vai empurrando este material até que este fique pronto. As vantagens deste modelo são: a redução dos níveis de estoque de matéria-prima, pois somente material necessário é comprado, e aumento da eficiência produtiva. Uma desvantagem que pode ocorrer é um acúmulo de material em processo, pois conforme a matéria-prima vai sendo empurrada, esta deve entrar no próximo ponto de processamento, e se este não estiver disponível, um estoque intermediário é gerado.

3.4.3 Sistema Puxar

Esta técnica visa exatamente atacar a desvantagem do modelo anterior. A produção só é iniciada se houver uma necessidade na etapa seguinte da produção. Isto implica em reduções drásticas dos níveis de estoque em processo, e consequentemente dos *leads times* produtivos. Este tipo de planejamento e controle de produção é conhecido como *Just In Time* ou *JIT*. Devido à característica de atender seu sucessor, o sistema deve funcionar como uma engrenagem bem justa, e as etapas produtivas devem chegar no momento exato (*just in time*). Nada pode acontecer senão atrasos ocorrerão. Portanto quebras, defeitos e paradas na produção são inaceitáveis. A grande desvantagem deste modelo reside neste aspecto do perfeccionismo, pois a implantação de um modelo deste requer doutrina e atitude preventiva.

3.4.4 Manufatura Síncrona

Este modelo é o que é abordado neste trabalho. Devido a sua relevância, o próximo capítulo é dedicado a descrevê-lo. Em resumo, ele prega que a empresa deve indentificar os gargalos produtivos e sincronizar sua produção conforme o ritmo que este dita.

4 Teoria das Restrições

4.1 Introdução a TOC

No início da década de 70 o doutor Eliyahu Goldratt, físico israelense, desenvolveu uma formulação matemática para planejar a fábrica de um amigo. Esta formulação, segundo cita (Guerreiro, 1999), estava baseada em uma técnica de previsão do comportamento de um átomo. Esta técnica o auxiliou a tratar do problema com grande número de variáveis. E exatamente para diminuir o número de variáveis do modelo, Goldratt passou a se preocupar não com todo o sistema, mas sim com os seus gargalos. Gargalos produtivos, para ele, eram os recursos que possuíam limitada capacidade produtiva e, portanto impediam a empresa de produzir mais. Já no final da década de 70, fundamentado nestas idéias foi desenvolvido um sistema denominado *OPT - Optimized Production Technology* ou tecnologia otimizada da produção.

Para comercializar o produto criou-se uma empresa chamada *Creative Output Inc.* Com o desenvolvimento do software, Goldratt foi adaptando e aprimorando seus conceitos e identificando princípios que podiam ser tratados através de sua abordagem. Isto não foi um modelo teórico, mas sim um modelo obtido através da experiência em implantações do sistema. Um dos resultados deste trabalho foi o desenvolvimento do pensamento *OPT*, que consiste em uma série de princípios aplicados à produção.

Os conceitos aplicados na *OPT* podiam ser estendidos a toda empresa. A esta extensão deu-se o nome de Teoria das Restrições - *TOC (Theory of Constraints)*. A diferença chave entre *OPT* e *TOC* é gargalo e restrição. Enquanto gargalo é algo produtivo e totalmente voltado ao chão de fábrica, restrição é algo que impeça a empresa de atingir suas metas como, por exemplo, mercado, matéria-prima ou até mesmo um setor produtivo.

Assim, em meados da década de 80, Goldratt fundou o *Avraham Y. Goldratt Institute*. Vendeu o software a *Scheduling Technology Group Ltd. (www.stg.co.uk)*, de Londres, Inglaterra, com sede em Dallas, Texas e passou a se concentrar na difusão dos conceitos criados. Para isto, junto com Jeff Cox, escreveu o romance “A Meta”, que expõe seus conceitos através de Alex Rogo, personagem principal da obra, e Jonah seu conselheiro. Além destas obras, Goldratt escreveu os livros “Mais que Sorte”, “A Síndrome do Palheiro”, “Corrente Crítica” e “A Corrida pela Vantagem Competitiva”. Cada obra tem um enfoque didático que aborda um tema da *TOC*. Por exemplo,

“Corrente Crítica” aplica a *TOC* ao gerenciamento de projetos e “A Corrida pela Vantagem Competitiva”, aborda os conceitos aplicados à produção.

Para expandir suas idéias no Brasil foi fundado o Instituto Goldratt, tendo como fundadores o Sr. Miguel Abuhab e o Doutor Goldratt. O Senhor Abuhab, além do Instituto Goldratt, é fundador da empresa de softwares Datasul. Com esta aliança estratégica, as idéias da *TOC* estão, através do Instituto, sendo difundidas ao País, e através da Datasul, com o desenvolvimento do software *DBR*⁴, estão aplicando na prática a teoria nas empresas.

4.2 Filosofia de Pensamento

A seguir desenvolve-se um modelo para melhor compreensão dos princípios inerentes a filosofia de pensamento da *TOC*. Este modelo é constituído de um produto vendido C. Para fabricá-lo são necessários dois semi-acabados, A e B. Estes semi-acabados são derivados do processamento das matérias-primas A e B, pelas operações conforme indicado na figura 4.1. Estas operações não competem por recursos, isto é, cada operação é executada por um recurso distinto dos demais.

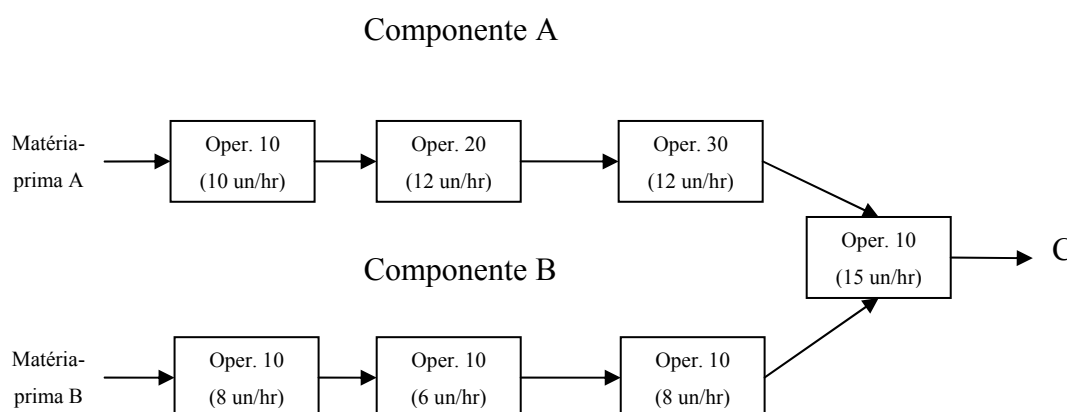


Figura 4.1 Modelo para fabricação do item C

Algumas questões a seguir são expostas para que seja possível acompanhar o raciocínio que conduz a filosofia *TOC*.

Quantos itens C podem ser produzidos? Ao se olhar para a linha de montagem do componente A, percebe-se que a limitação está na operação 10. Assim, o máximo produzido de A é 10 un/hr. Já para B, o limitante é a operação 50, com 6 un/hr.

⁴ Os algoritmos proprietários do software serão parcialmente exibidos neste trabalho

Assim, como para compor C, deve-se ter A e B juntos, e só se produz no máximo 6 un/hr, C tem uma taxa de produção de 6 un/hr. A operação 50 é a limitante do sistema como um todo e é considerada a **restrição**. Para se aumentar a eficiência de todo o sistema, não é necessário investir em melhorias de eficiência em todos os recursos, apenas na restrição.

O que acontece se a produtividade da operação 10 for aumentada, de 10 para 15 un/hr? Na concepção atual da administração da produção isto é o ideal, ainda mais se o investimento for baixo. Porém, em relação à produtividade do item C, nada é alterado. O que acontecerá é que agora o item A é produzido mais rapidamente e por consequência, na linha de montagem, haverá um acúmulo maior deste produto, pois a taxa de produção de C continua sendo de 6 un/hr.

E se a operação 40 não puder ser realizada? Neste caso, podem-se ter duas alternativas. A primeira é se existir estoque entre os dois processos, o processo sucessor (operação 50) o consumirá até que este se acabe ou que a operação 40 volte a ser realizada. Neste ponto, um dos conceitos da *TOC* aplicado à produção “entra em cena”: “o pulmão”. Deve-se citar neste ponto o problema que pode ocorrer em abordagens *JIT*, que tendem a eliminar os estoques em processos. Em caso de não haver estoque, a segunda alternativa vem à tona. A operação 50 simplesmente para e a produção de C é prejudicada.

A operação 60 ficará em manutenção por um turno. O que ocorre? A restrição, operação 50, continua produzindo enquanto houver espaço para gerar estoque. Em caso de não haver espaço disponível, novamente a restrição deverá parar e o sistema todo deixa de produzir o número de peças que a restrição ficou impossibilitada de gerar.

Após estes questionamentos, observa-se que não adianta tentar otimizar as partes de um sistema, o grande dilema aqui é: **os ótimos locais não conduzem aos ótimos globais**. Não adianta em nada as operações não restritivas produzirem ao seu máximo, porque estoque intermediário será gerado antes da restrição. E após não é possível produzir mais rápido do que a taxa de produção desta restrição.

4.2.1 Componentes da Teoria das Restrições

Apesar da *TOC* ter surgido devido a necessidades em sistemas manufatureiros, ela foi estendida e possui diversos componentes que permitem sua aplicação em qualquer processo decisório. A figura 4.2 extraída de (Cox e Spencer, 2002) esquematiza seus componentes.

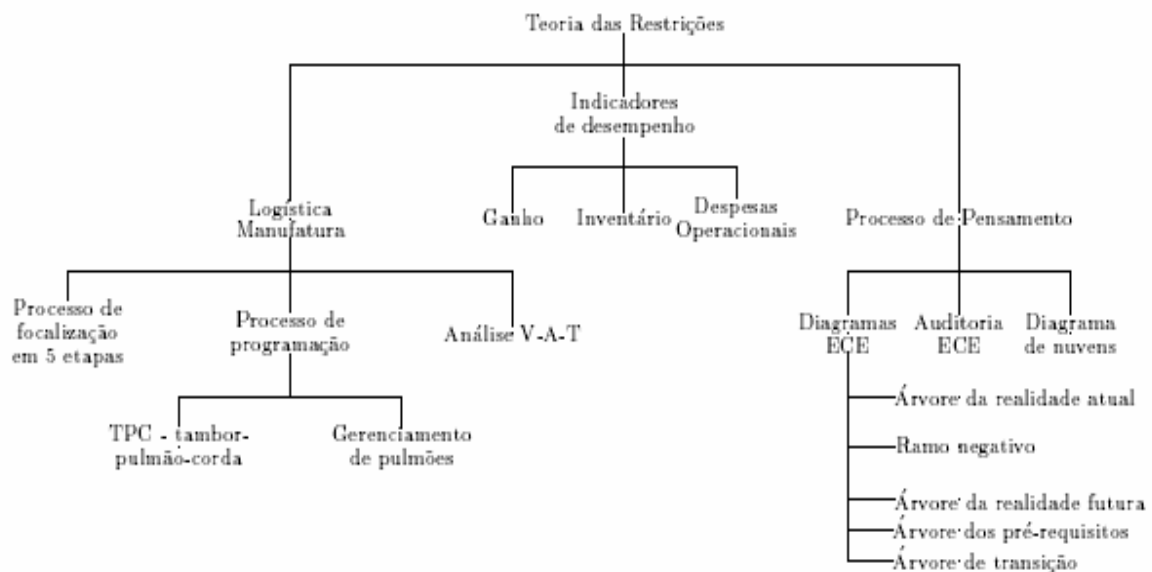


Figura 4.2 Componentes da Teoria das Restrições

O ramo logístico aborda questões produtivas, como o processo de focalização explicados em tópicos subsequentes, análise V-A-T, que permite classificar as empresas segundo seu modelo V (poucas matérias-primas e muitos itens acabados – petroquímicos); A (muitas matérias-primas e poucos itens acabados – eletrodomésticos) e T (um híbrido entre os dois) e adaptar a teoria a suas necessidades. Ainda possui o processo de programação da produção baseado nos conceitos de tambor, pulmão e corda, conhecido como TPC. Em inglês, usa-se a sigla *DBR* (ver glossário). O processo de programação também possui o gerenciamento de pulmões que permite ao usuário definir qual o tamanho dos pulmões e seu conteúdo, possibilitando um melhor controle do chão de fábrica.

Um ramo de indicadores, necessários para se medir o ganho em sistemas baseados na TOC, discutidos em seção subsequente e por fim o ramo decisório (exposto na figura como processo de pensamento) que consiste em um conjunto de ferramentas para auxiliar na tomada de decisão, identificar os problemas e possíveis soluções.

4.3 Etapas de Focalização da TOC

A TOC prega que a preocupação deve ser voltada às restrições que impedem a empresa de atingir suas metas. Para isto, foi desenvolvido um processo de focalização de cinco etapas. Para ilustrar estas etapas, considera-se a seguir uma empresa com linha dedicada, que produz apenas o item C. A figura 4.3 exibe o processamento necessário para montar uma unidade de C. Os valores nas operações correspondem à capacidade

máxima do setor. O valor ao lado do item na expedição (300 uns/hr) refere-se ao valor máximo que o mercado está disposto a comprar. A empresa trabalha oito horas por dia e tem uma política de realizar no máximo 10% de horas extras ao mês. Partindo-se deste cenário, são aplicados os cinco passos.

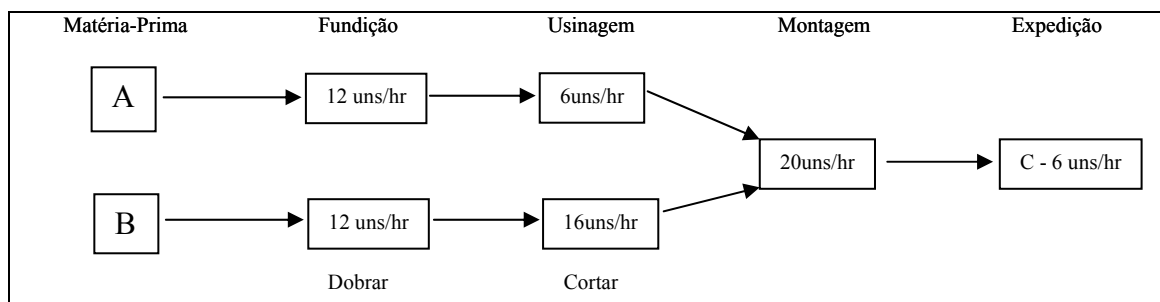


Figura 4.3 Esquema de produção do produto C

4.3.1 Identificação da Restrição

Restrição é algo que impede a empresa de atingir a meta. Definindo-se a meta como lucro, a restrição é o que impede a empresa de obtê-lo. E em sua maioria, as restrições são enquadradas como físicas ou políticas. Restrições políticas são, por exemplo, o número máximo de horas extras permitido, políticas de compra, venda e lotes de fabricação. Restrições físicas são, em geral, mais visíveis, pois são facilmente identificáveis no processo produtivo. Um forno ou uma extrusora para fabricação de bobinas pode ser uma restrição física. Para facilitar o entendimento, a tabela 4.1, exhibe a capacidade semanal de produção de cada área.

Tabela 4.1 Tabela de análise da capacidade da empresa

Área	Capacidade	Hrs/Dia	Dias/Sem	Hrs/Semana	Total un/sem
Fundição	12	8	5	40	480
Usinagem	6	8	5	40	240
Montagem	20	8	5	40	800
Dobragem	12	8	5	40	480
Cortagem	16	8	5	40	640
Mercado					300

Através do exemplo, pode-se verificar que o setor que limita a produção total da empresa é o setor de usinagem. Com uma capacidade de 240 uns/semana, não é possível, sem alguma ação, produzir mais que este valor. O próximo passo trata de agir sobre a restrição.

4.3.2 Explorar a Restrição

O objetivo é aumentar a produtividade da empresa. A área limitante é a usinagem. Para aumentar a produtividade, deve-se identificar o que é possível fazer para aumentar sua capacidade. Algumas alternativas poderiam ser:

- ☒ Aplicar horas extras;
- ☒ Resequenciar as ordens para minimizar tempos de preparação de máquinas;
- ☒ Terceirizar serviços;
- ☒ Investir em máquina, equipamento ou pessoal neste setor.

Por suposição, aplica-se horas extras ao recurso restritivo. Para facilitar o raciocínio, admita-se um aumento de 10% na produtividade. Isto provoca um aumento de 240 para 264 uns/semana. A conclusão imediata é que ao se aplicar este aumento de horas extras apenas no recurso restritivo, permitiu-se um aumento em toda a produtividade da empresa. Mesmo que nos demais setores não sejam aplicados às horas extras.

É possível ainda tentar diminuir os tempos de preparação ou reativar alguma máquina antiga para realizar estas tarefas. Ou ainda comprar novos equipamentos, contratar pessoal ou terceirizar algumas atividades neste setor.

Depois de realizada a análise das alternativas para explorar a restrição, deve-se garantir que esta não fique sem material para processar. Desta necessidade deriva-se o próximo passo.

4.3.3 Subordinar tudo à restrição

No passo anterior definiu-se o que será realizado pelo recurso restritivo. Esta fase consiste então em, comunicar aos demais recursos o que será realizado pelo recurso restritivo e determinar o que e quando os outros recursos devem iniciar suas operações, para garantir o processamento pela restrição.

Como os demais recursos possuem capacidade excedente em relação ao recurso restritivo, isto pode implicar em horas ociosas, paradas ou folgas. Na atual concepção da administração da produção isto é inaceitável. Esta talvez seja a fase mais difícil de conceber, pois deixar funcionários ociosos enquanto outros (os do *RRC*) ficam 100% em atividade não é algo comum.

E se ações forem tomadas para aumentar definitivamente a capacidade desta área? A próxima fase descreve esta ação.

4.3.4 Elevar a restrição

Após um período de análise, os gerentes decidiram comprar mais um centro de usinagem, dobrando sua capacidade. A nova distribuição da empresa fica conforme a tabela 4.2.

Tabela 4.2 Tabela de análise da capacidade da empresa após aumento de capacidade

Área	Capacidade	Hrs/Dia	Dias/Sem	Hrs/Semana	Total
Fundição	12	8	5	40	480
Usinagem	12	8	5	40	480
Montagem	20	8	5	40	800
Dobragem	12	8	5	40	480
Cortagem	16	8	5	40	640
Mercado					300

Com o investimento realizado, a capacidade do setor de usinagem dobrou, porém a da empresa não. Isto porque a restrição mudou, e agora o limitante é o mercado. Entenda-se por mercado, o quanto é possível vender. A empresa agora tem capacidade de prover toda a demanda de mercado, mas o mercado não absorve. O próximo passo descreve o que fazer.

4.3.5 Não deixar a inércia tomar conta, e voltar ao passo um

Neste procedimento, é possível que os administradores deduzam que resolveram todos os problemas. Mas isto não é verdade, pois é possível verificar que ocorreu um deslocamento da restrição para outro setor. Não há mais restrições físicas na empresa. A restrição deslocou-se para o mercado. As vendas agora são os limitantes da empresa.

Os esforços devem ser canalizados para esta área agora. Todo o processo deve ser iniciado novamente, voltar ao passo um (identificação da restrição) é a recomendação. Isto implica em um processo de melhoria contínua.

Porém é preciso mudar a forma de se medir o desempenho da empresa. A seção seguinte esboça os indicadores de desempenho da *TOC*.

4.4 Indicadores de Desempenho da TOC

A grande mudança neste ponto trata de visualizar, não mais os custos, mas sim os ganhos. Muda-se da contabilidade de custos para a contabilidade de ganhos.

Conforme analisado por (Cox e Spencer, 2002) é possível manter a análise tradicional, para efeitos governamentais e para as políticas empresariais utilizar a análise de ganhos, principalmente nos dias informatizados como os de hoje. A visão muda radicalmente. Não existe mais a preocupação com o rateio dos custos. E a análise, na contabilidade de ganhos é realizada pelo ganho na restrição.

A *TOC* define três indicadores globais. Um é absoluto - o “lucro líquido”. O outro é relativo, chamado de “retorno sobre o investimento”. O terceiro e último é o “fluxo de caixa”, um indicador necessário. Para entender a relação entre estes indicadores, na visão da *TOC*, o primeiro expressa o quanto de lucro em dinheiro a empresa está gerando. Por exemplo, uma empresa auferiu cem milhões de lucro em 2004. A pergunta seguinte é se isto é bom ou ruim. Se o investimento foi de cinquenta milhões isto é bom, porém se foi de um bilhão, sua significância é questionável.

Desta forma o segundo indicador visa mensurar o quanto representa o primeiro indicador em relação ao investimento realizado. Para o primeiro caso se teria um *ROI* de seis meses enquanto que no segundo, este representa dez anos, supondo-se nulas a taxa de juro anual.

O terceiro indicador é necessário para mensurar a “saúde” da empresa.

Estes indicadores, porém, não podem ser medidos diretamente pela produção. Há a necessidade de ligá-los diretamente ao chão de fábrica. O próximo passo é desenvolver indicadores que substituam a eficiência dos operadores, setores ou máquinas por mensuradores da *TOC*. Assim, Goldratt desenvolveu os indicadores “ganho”, “inventário” e “despesa operacional”. A seguir é comentado cada um.

4.4.1 Ganho ou *Throughput*

É definido, conforme (Goldratt e Fox, 1992), como o índice pelo qual o sistema gera dinheiro através das vendas. Este, segundo a *TOC*, é o preço de venda dos produtos menos o que se paga aos fornecedores, vendedores, governo, transporte. Deve-se observar que neste momento o custo produtivo não é levado em conta. É simplesmente venda menos custo direto. E este só pode ser medido após entregue o produto ao cliente. Estoque neste ponto não é considerado ganho. Para isto define-se outro indicador: o inventário.

4.4.2 Inventário

É definido, segundo (Goldratt e Fox, 1992), como todo o dinheiro que o sistema investe na compra de coisas que o sistema pretende vender. Ou de outra maneira, conforme (Cox e Spencer, 2002) apud (Cox III e Blackstone, 1998) "*são todos os itens comprados que podem ser revendidos e incluem bens acabados, estoques em processo e matérias-primas. O inventário é sempre avaliado pelo valor de compra e não inclui qualquer tipo de agregação de valor, fazendo oposição às práticas tradicionais de contabilidade de custos de agregar mão-de-obra direta e alocar custos indiretos conforme o material em produção evolui através do processo de produção*". Fica clara a diferença dada aqui ao inventário. Não agregar qualquer custo que não seja o valor pago. Qualquer outro valor adicional deve ser incluso na despesa operacional.

4.4.3 Despesa Operacional

Por fim, o último indicador é definido, novamente segundo seu criador (Goldratt e Fox, 1992), como todo o dinheiro que o sistema gasta para transformar inventário em ganho. Aqui deve-se incluir os gastos administrativos, mão-de-obra direta, energia elétrica, aluguéis, manutenção de máquinas e qualquer outro item não contemplado no inventário.

Estes indicadores são essenciais para mover a produção em direção à meta. A figura 4.4 exhibe como estes indicadores se relacionam com os indicadores globais da empresa. Deve-se enfatizar que um é relacionado à meta da empresa enquanto o outro é relacionado ao operacional, mas focado a indicadores internos ou de produção.

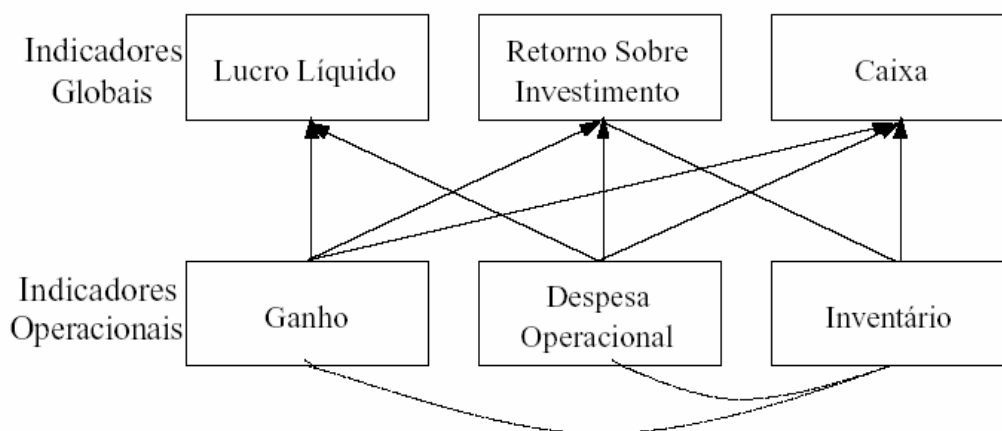


Figura 4.4 Relacionamento entre indicadores globais e de produção

Os relacionamentos expostos na figura acima exibem que não se pode avaliar os indicadores da *TOC* de forma isolada. Isto porque estes possuem relacionamento com os demais. Por exemplo, ao se reduzir o inventário, diminui-se o capital investido em estoque. Diminuindo o valor investido em estoque, diminui o capital aplicado em matéria-prima, que por sua vez impacta no fluxo de caixa. Reduzindo o fluxo de caixa e mantendo-se o mesmo ganho, aumenta-se o lucro. Análises similares podem ser feitas com os demais indicadores.

A próxima seção trata de descrever como a produção deve agir para melhorar estes indicadores.

4.5 Programação da Produção

A última abordagem a ser analisada neste capítulo é a programação da produção. As técnicas utilizadas pela *TOC* podem ser aplicadas juntamente com outras técnicas, conforme discutido no capítulo anterior. O Plano Mestre, o *MRP* e o *JIT* são utilizados de forma híbrida para se atingir a meta global. A filosofia para a produção consiste, basicamente, na aplicação dos conceitos conhecidos como Tambor-Pulmão-Corda ou TPC. Em inglês a terminologia é conhecida como *Drum-Buffer-Rope* ou *DBR*. Estes conceitos derivam da aplicação do processo de focalização de 5 etapas. No capítulo 6, descreve-se sucintamente cada fase. Aqui se procura definir os conceitos de tambor, pulmão e corda.

4.5.1 Tambor

Conceito derivado da comparação com uma “tropa de escoteiros ou de militares”, o qual é ritmada pelo bumbo do tambor. Este determina como a produção deve responder, definindo “quando o material deve estar chegando” e “quando o material deve ser entregue”. Para identificá-lo deve levantar qual recurso que é gargalo produtivo, conforme visto no tópico 4.3.1. A figura 4.5 exibe uma linha de produção, de forma que cada círculo representa uma operação. Os valores contidos em cada uma equivalem à capacidade de produção em uns/hr. O tambor em destaque é o recurso com a menor capacidade, o qual dita a produção da fábrica.

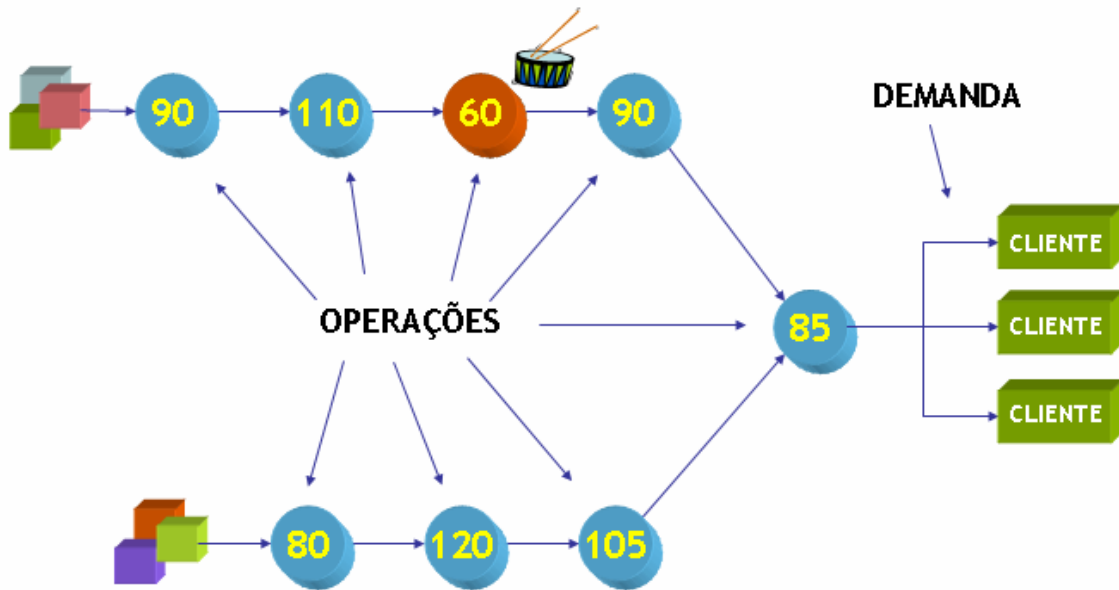


Figura 4.5 Identificação do tambor ou gargalo produtivo

4.5.2 Pulmão

As dificuldades em um ambiente fabril são inúmeras e imprevistos são inevitáveis. A matéria-prima pode atrasar, máquinas podem parar ou trabalhadores podem faltar. Para evitar que a restrição pare, insere-se um “pulmão”. “Pulmão” é uma proteção de tempo inserido antes do gargalo produtivo, linhas de montagem e expedição para evitar atrasos. É uma antecipação imposta às operações precedentes à operação restritiva. Impõe-se à fábrica que entregue tudo com um “pulmão” de antecedência (chamado de pulmão de expedição). Desta forma; garante-se que qualquer problema que venha a ocorrer no chão de fábrica, não impeça a restrição de produzir, já que esta começa a consumir o material do pulmão. Este pulmão é a garantia que se dá contra as flutuações estatísticas inerentes a todo processo. A figura 4.6 exibe os pulmões antes da restrição, de convergência e antes da expedição (mercado).

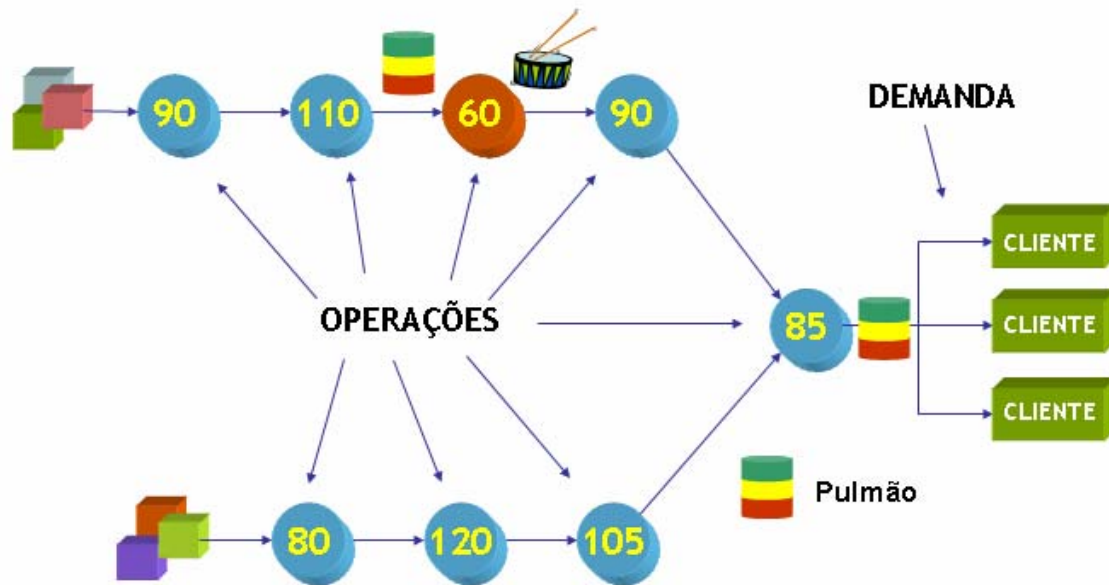


Figura 4.6 Inserção dos pulmões de restrição, convergência e expedição

4.5.3 Corda

Por fim, "liga-se" à programação do tambor a liberação de matéria-prima. A esta ligação dá-se o nome de "corda". A "corda" outorga as datas em que a matéria-prima deve ser liberada para que se evite a parada da restrição. Sua função é sincronizar os recursos não restritivos em função das necessidades dos pulmões. Lembrando que se a matéria-prima for liberada antes, ficará aguardando antes da restrição por um período similar à antecipação. Se for liberada depois, pode implicar em parada da restrição. A figura 4.7 exibe uma ilustração das cordas, ligação da liberação de matéria-prima com o pulmão.

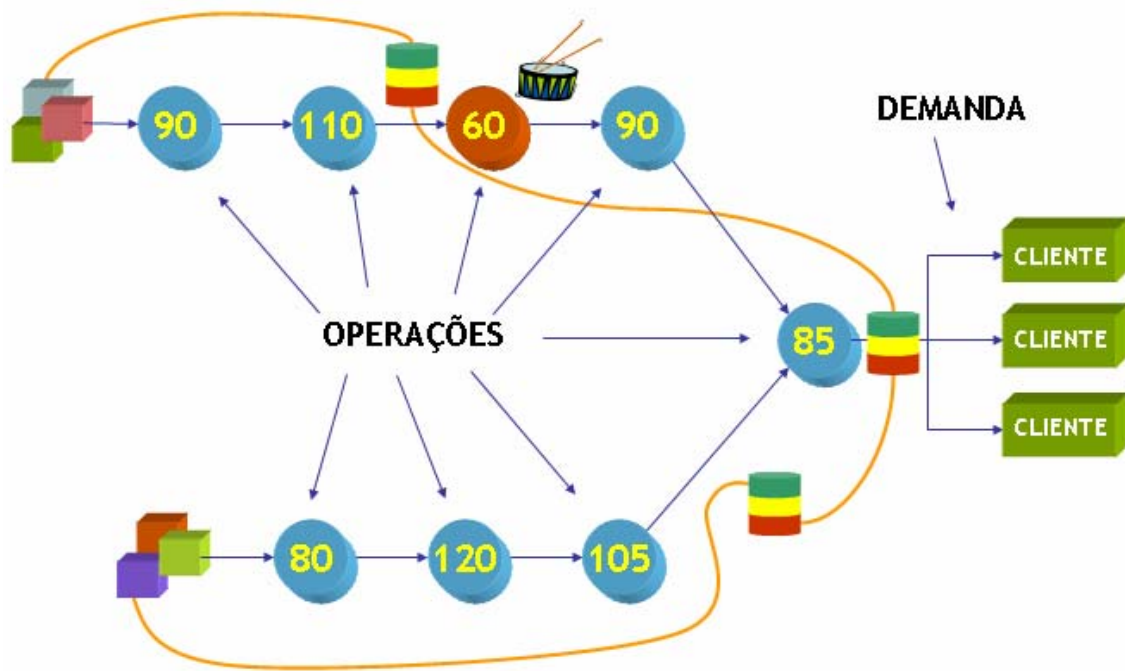


Figura 4.7 Ilustração das “cordas” conectando os “pulmões”

Esta abordagem parece simples, mas a realidade do chão de fábrica impõe restrições complicadoras. Objetivos não bem claros e estratégias diferentes na estrutura da empresa implicam em um quadro complicado. Programar a produção em uma empresa não significa programar gargalo, porém, programar gargalo implica em aumentar o ganho da empresa.

No próximo capítulo detalha-se o papel do sequenciamento enquanto que o capítulo 6 descreve um método para se programar a produção.

5 Sequenciamento

O problema de sequenciamento é referenciado na literatura em língua estrangeira, diga-se de língua inglesa, como *scheduling*. Adequando-se sua semântica a compreensão dos problemas de scheduling, o verbete “programar” reflete de maneira mais adaptada, contudo o vocábulo sequenciamento é utilizado mais amplamente na literatura em língua portuguesa para designar estes problemas. Para compreender a diferença entre estes termos deve-se observar que “sequenciar” consiste em definir a ordem em que tarefas, operações ou atividades devem ser realizadas, enquanto que “programar” consiste em alocar recursos ao longo do tempo para realizar um conjunto de tarefas. Percebe-se que o termo “programar” é mais abrangente que sequenciar, ou seja, o termo, “sequenciar”, está implícito no primeiro. Para determinar quando um recurso será alocado para uma dada atividade, deve-se determinar primeiro a ordem em que estas serão feitas e subsequentemente calcular a sua data de início. O termo sequenciamento será usado indistintamente, tanto para programação quanto para sequenciamento, cabendo ao contexto a identificação do sentido do vocábulo.

5.1 Introdução

Sequenciamento consiste na alocação de recursos ao longo do tempo para realizar um conjunto de tarefas. Os recursos e tarefas podem assumir diversas formas. Os recursos podem ser máquinas em um ambiente de trabalho, pessoas em um projeto, pistas de um aeroporto, o processador em um computador, entre outros. As tarefas podem ser as operações a serem executadas nas máquinas de um ambiente de trabalho ou projeto, os algoritmos alocados pela CPU ou os aviões em processo de decolagem ou aterrissagem. Cada uma destas tarefas possui uma duração, uma data mais cedo, uma prioridade e uma data de entrega. Os objetivos podem ser: entregar as tarefas o mais cedo possível, ou ainda, entregar o maior número de tarefas no prazo.

Sequenciamento, conforme (Baker, 1974), é um processo decisório, uma teoria que possui seus princípios, modelos, técnicas e conclusões lógicas. Possui terminologia e notação particulares. Sua teoria pode ser estendida e aplicada a outros processos decisórios, que possuam características similares. Estes processos decisórios são estudados em otimização combinatória e, em particular, encontra-se amparo na teoria dos grafos.

O sequenciamento ou função sequenciamento encontra-se presente em diversos problemas decisórios. Entretanto o principal interesse do capítulo é tratar dos problemas de sequenciamento voltados ao planejamento produtivo.

Na produção, a função sequenciamento é uma decisão secundária. Primeiro é necessário responder a algumas perguntas, como: “o que deve ser feito”, “quanto” e “para quando”. A figura 5.1 reexibe a figura 2.1 e indica a localização da função sequenciamento dentro do contexto do planejamento. Encontra-se dentro do planejamento de curto prazo, e é apenas um de seus componentes. Sua importância está em definir a sequência de tarefas realizadas por um determinado recurso objetivando-se (i) utilizar os recursos eficientemente, (ii) dar rápidas respostas ao mercado e (iii) respeitar os prazos de entrega.

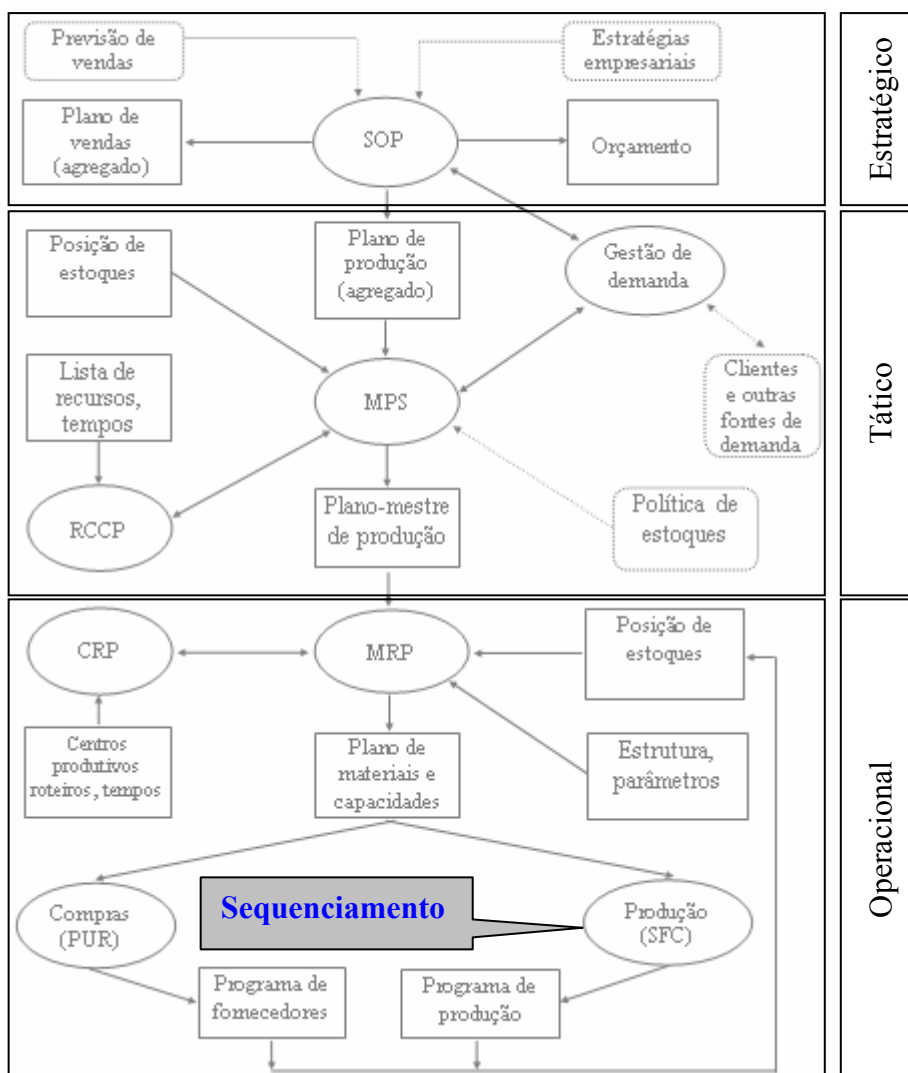


Figura 5.1 Localização da função sequenciamento dentro do planejamento da produção

Para entender a dificuldade em sequenciar ordens de produção, deve-se imaginar a realidade dentro do processo produtivo. O ciclo começa com a definição “do que deve ser produzido”, etapa em que as ordens de produção são geradas. As ordens carregam as informações “do que produzir” e “para quando”. Cada ordem possui uma estrutura, que identifica o que é necessário para se produzir o item solicitado. Ao ato de verificar os componentes necessários, chama-se explosão. Então a próxima etapa é a realização da explosão pelo *MRP*. O *MRP* determina tudo “o que é necessário produzir” e “para quando” em níveis mais baixos. Isto posto, tem-se a imposição de precedências dada pela estrutura do produto que está vinculado a ordem, que deve ser respeitada. Tendo todas estas informações em mãos, a dinâmica mercadológica impõe atrasos de fabricação devido a ordens de maior prioridade estarem ocupando os postos de trabalho. Além do mais, existe a possibilidade das máquinas ou equipamentos quebrarem ou estarem em manutenção. Os tempos previstos para processamento, por vezes, não correspondem à realidade, atrasando as demais tarefas a serem processadas. Outros problemas como atraso de fornecedor implicam em falta de material na hora necessária e consequentemente atrasos produtivos.

É claro que um sistema não consegue contemplar por completo esta realidade. Contudo os modelos desenvolvidos devem capturar as essências dos problemas de sequenciamento reais. Isto fez com que nas últimas décadas este problema ganhasse importância na academia. Nas seções subsequentes a função sequenciamento é descrita, voltada ao processo produtivo.

5.2 Definição

Sequenciar consiste em determinar a ordem em que operações são executadas, definindo os recursos a serem utilizados e datas de início. Formalmente, o problema é representado pelo símbolo Π_j , consistindo em um conjunto de tarefas \mathcal{T} , geralmente finito, que deve ser processado por um conjunto de recursos \mathcal{M} , também considerado finito. As tarefas consistem em um conjunto de operações \mathcal{O} , normalmente possuindo uma ordem pré-determinada, chamada restrições de precedência, que devem ser processadas nos recursos. Existem diversos termos e considerações que implicam em diferenciação no tratamento do sequenciamento. Para isto, a seção seguinte aborda a notação comumente adotada.

5.3 Notação

A notação adotada para Π_j segue o padrão exposto em (Pinedo, 2002). Este padrão consiste em uma tripla $\alpha \mid \beta \mid \gamma$ que representa respectivamente o ambiente de recursos, detalhes sobre o processamento e restrições adicionais e a função objetivo. Porém, não é a única existente. Em (Jain, 1988) tem-se uma breve abordagem sobre as notações de Conway, Maxwell e Miller de 1967, consistindo de uma 4-tupla $A \mid B \mid C \mid D$. Ainda em (Jain, 1988), cita-se as modificações propostas por MacCarthy e Liu, 1993 no elemento C. A notação $\alpha \mid \beta \mid \gamma$ foi proposta por Graham, Lawler, Lenstra and Rinnooy Kan (1979) e alterada por Pinedo em seu livro (Pinedo, 2002).

5.4 Terminologia

Para o entendimento do que cada campo representa é necessário definir algumas variáveis adicionais. Estas variáveis são: tempo de processamento, data mínima, data de entrega e prioridade. A seguir comenta-se brevemente o significado de cada uma.

Tempo de Processamento (*processing time*) p_{ij} : Tempo necessário para processar a operação i no recurso j . Em caso de apenas um recurso, ou de recursos idênticos, omite-se o índice j .

Data prometida (*due date*) d_i : Data em que a operação deve estar pronta, pois a partir de acordos comerciais, clientes, entre outros, pode incidir multas ou até perda do cliente, com o não cumprimento desta data. Ela determina a partir de que instante que se começa a contar o atraso. Em caso de recursos com calendários diferentes, acrescenta-se o índice j .

Prioridade (*weight*) w_i : Denota a importância de uma determinada tarefa. Esta prioridade, também denotada por peso, pode representar diversos fatores, como custos de atraso, custos de fabricação de um produto, ganho, tipo de tarefa, cliente, ou qualquer fator relevante ao processo decisório.

Data de término (*completion time*) c_i : Variável de saída do problema. Determina o instante que a operação se encerra. Também denominada de instante de término. É obtida somando-se o tempo de processamento p_{ij} a data de início s_i .

5.4.1 O Parâmetro α

Este parâmetro informa o ambiente operacional. Detalhes de quantidade de máquinas, tipos de máquinas e características produtivas. O conjunto de valores que o

parâmetro α pode assumir é $\{1, Pm, Qm, Rm, Fm, FFc, Jm, FJc, Om\}$. Cada parâmetro é descrito a seguir.

Uma máquina (*single machine*) (I): É o mais simples dos modelos, onde se tem apenas um recurso ou máquina.

Máquinas idênticas em paralelo (*Pm*): Existe um grupo de máquinas que possuem as mesmas características e podem processar uma determinada operação. Um exemplo seria um grupo de dobradeiras, as quais possuem as mesmas características. Em caso da operação permitir apenas que um subconjunto a processe, deve-se adicionar o elemento M_j no parâmetro β .

Máquinas em paralelo com diferentes velocidades (*Qm*): Existe um grupo de máquinas, que realizam a mesma tarefa, porém possuem *performances* diferenciadas. Isto pode ser entendido por um grupo de tornos manuais e automáticos. Ambos realizam a mesma atividade, porém o automático a realiza com uma velocidade maior. A velocidade da máquina j é denotada por v_j . O tempo p_{ij} que a tarefa i leva para ser processada na máquina j é p_i/v_j . Em caso de v_j ser igual a 1 para todas as máquinas, o modelo se reduz ao anterior (*Pm*).

Máquinas em paralelo não relacionadas (*Rm*): É a generalização do modelo anterior (*Qm*). Cada operação possui características peculiares em cada centro, e a velocidade é dependente da tarefa v_{ij} . O tempo para processar a operação i é, p_i/v_{ij} . Caso as velocidades dos recursos sejam a mesma para todas as tarefas, $v_{ij} = v_j$, retorna-se ao modelo anterior (*Qm*).

Linha de Produção - *Flow Shop* (*Fm*): Conhecido como linha de produção, têm-se n máquinas em série e que cada tarefa deve passar na sequencia de máquinas predeterminadas. A decisão é definir a ordem das tarefas, sendo que em cada máquina é criada uma fila. Se não for possível uma operação, de uma determinada tarefa, em uma outra máquina, passar a frente de uma operação de uma outra tarefa, isto implica que não se pode trocar a ordem de entrada das tarefas. Nesta situação o problema recebe no campo β o valor *prmu* que vem permutação.

Linha de Produção Flexível - *Flexible Flow Shop* (*FFc*): Neste modelo existem estágios pré-determinados, e cada estágio possui máquinas em paralelo, também chamados grupos de máquinas. Desta forma, este modelo agrega um dos tipos do conjunto $\{Rm, Qm, Pm\}$ e *Fm*. É uma generalização dos anteriores, de forma que a

política de “primeiro a entrar, primeiro a sair” pode ou não ser utilizada. Cada tarefa passa pelo grupo de máquina 1, grupo 2 e assim sucessivamente.

Oficinas de Tarefas - Job Shop (Jm): Este é o ambiente mais comum nas empresas. Consiste em diversas máquinas especializadas e tarefas que possuem um roteiro próprio. As diferenças podem ser visualizadas nas figura 3.2 e figura 3.3 apresentadas na seção 3.2.2. Geralmente uma tarefa é processada uma única vez em um determinado grupo de máquina. Em caso de uma tarefa retornar ao mesmo grupo dá-se o nome de recirculação, *recrc* é incluído no campo β .

Job Shop Flexível (FJc): Similarmente a *FFc*, adiciona-se grupos de máquinas ou máquinas em paralelo. Cada tarefa possui seu próprio roteiro e, se é possível passar em um grupo novamente, adiciona-se *recrc* no campo β .

Open Shop (Om): As tarefas devem ser processadas em cada máquina, porém o tempo de processamento pode ser zero. Os roteiros podem ser distintos para cada tarefa e podem ser definidos pelo sistema.

5.4.2 O parâmetro β

Este parâmetro informa restrições do ambiente operacional. Aqui é apresentada uma lista, não exaustiva, dos possíveis campos. São incluídos neste trabalho dois novos campos: ferramentais produtivos (*tool*) e calendário de trabalho (*routine*, *rtn*), não encontrado na literatura consultada. O conjunto de valores analisados é $\{r_j, s_{jk}, prmp, prec, brkdown, M_j, prmu, block, nwt, recrc, tool, rtn\}$. Cada elemento deste conjunto é brevemente explicado abaixo.

Data Mínima (ready ou release date) dm_{ij} ou r_i : data que determina a partir de que instante uma operação pode ser processada. Isto se deve ao fato de, por exemplo, uma operação não estar pronta para processamento no instante 0 devido à falta de material, ou processos antecessores.

Tempos de Preparação (setup times) - s_{ik} : A variável s_{ik} representa o tempo necessário para preparar um recurso, sendo que a operação processada foi a i e a que será processada é k . Como exemplo, empresa que trabalha com cores, preparar uma máquina da cor branca para a azul requer pouca limpeza, porém o inverso, do azul para o branco, requer a limpeza total do recurso. Se o tempo de preparação é dependente da máquina, tem-se s_{ijk} .

Preempção (prmp): Este parâmetro informa ao modelo que é permitido interromper uma tarefa durante seu processamento e reiniciá-la mais tarde, do ponto em

que parou. Uma aplicação deste modelo está nos sistemas operacionais dos computadores. Uma aplicação pode ser preemptada (interrompida) para que outra assuma o processador. Em situações reais, não é comum a preempção, isto porque as características produtivas, como citadas nos capítulos anteriores, conduzem a sistemas de produção em lotes. A preempção é utilizada em casos excepcionais, por exemplo, quando uma fábrica troca a prioridade das operações, e interrompe sua produção para atender a um cliente específico. Mas este tratamento não é realizado em nível de modelo.

Restrições tecnológicas - (prec): Quando, por implicações estruturais e de processamento, uma tarefa possui roteiros de fabricação, e estes roteiros exigem que uma operação deve ser executada antes de outra, temos restrições de precedência ou tecnológicas. Este é o caso da maioria das empresas, por exemplo, para a montagem de um item é necessário produzir seus subitens, portanto, a operação de montagem só pode iniciar após a produção de seus componentes.

Paradas de máquina ou *breakdown* (brkdown): Em caso de diferentes disponibilidades de máquinas, ou diferentes calendários de trabalho, as máquinas ou recursos não ficam disponíveis todo o tempo. Desta forma o número de máquinas permissíveis para alocação é dependente do tempo $m(t)$. Este é o caso de empresas que trabalham com turnos diferentes em seus setores. Empresas que realizam com frequência a manutenção de equipamentos e quando máquinas estão sujeitas a paradas.

Recursos válidos - M_j : Quando se possui um ambiente do tipo Pm , Rm ou Qm , é possível que nem todas as operações possam ser processadas por todas as máquinas do grupo. Neste caso é necessário listar quais as máquinas estão aptas a receber a operação. Como exemplo, empresas que possuem prensas com moldes pré-instalados, não aceitam qualquer tarefa, apenas as que exigem o mesmo molde.

Permutação (prmp): Conforme citado em FM e FFC , é aplicado quando a ordem de entrada não pode ser alterada. O modelo é primeiro a entrar primeiro a sair, conhecido como *FIFO* (*First In First Out*).

Sem Espera (No-wait)(nwt): Acontece quando uma tarefa não pode esperar em nenhum ponto do sistema. Uma vez que se tenha iniciado seu processamento, todas as operações devem ser executadas imediatamente. Este é um exemplo de linhas de produção de aço. Uma vez o produto tenha iniciado na linha de fusão, deve percorrer todo o processo.

Recirculação - (*recrc*) : Se uma tarefa passar duas vezes por um mesmo recurso ou grupo de máquinas, dá-se o nome de recirculação.

Ferramentas válidas (*tool*) - F_j : Da mesma maneira que é possível haver centros válidos, há a possibilidade de se possuir ferramentas válidas. Esta restrição implica que não só o centro deve estar disponível para o processamento, bem como a ferramenta. Para ambientes com uma máquina, este caso não tem efeito, apenas em ambientes com recursos paralelos (Pm , Rm e Qm).

Neste tipo de problema é necessário designar a ferramenta e o centro para a operação, consistindo em um complicador.

Existem outros elementos permissíveis neste parâmetro, mas são auto-explicativos.

5.4.3 O parâmetro γ

Para problemas de sequenciamento, os objetivos são em função de sua data de término c_i , ou seja, $f(c_i)$. Em essência γ pertence ao conjunto $\sum \{f(c_i), f_{\max}(c_i)\}$. De forma que:

- ☑ $\sum f(c_i)$: Representa minimizar o somatório da função f para cada tarefa ou operação. Ela é representada por:

$$\min(f(c_1) + f(c_2) + \dots + f(c_n)) \quad (5.1)$$

Onde min significa minimizar.

- ☑ $f_{\max}(c_1)$: Representa minimizar o maior valor entre os valores de f . Sua definição é:

$$\min \max(f(c_1), f(c_2), \dots, f(c_n)) \quad (5.2)$$

Onde max representa máximo e min max significa minimizar o máximo.

Para cada função acima, é possível incluir um peso w , de forma a refletir a importância de uma tarefa ou operação. A importância é devido a diferenciações naturais que são inerentes ao processo produtivo. Um cliente internacional, que depende da data de embarque no navio, tem prioridade maior do que um nacional. Uma operação que já se encontra em produção, ao alimentar o sistema, de alguma forma deve-se informar que este deve ser processado antes. O peso tem a finalidade de

incorporar estas necessidades. As funções para γ pertencem ao conjunto generalizado $\sum \{f(w_i, c_i), f_{\max}(w_i, c_i)\}$. Se w_i for constante e igual a 1, tem-se o conjunto anterior.

As funções de otimização f mais utilizadas e referenciadas na literatura estão listadas a seguir. A seguinte sintaxe é utilizada (**Tradução/Termo Original/Sigla**). Para uma referência mais completa consultar (Pinedo, 2002).

Cobertura/Makespan/ C_{\max} : Definido como

$$\max(C_1, C_2, \dots, C_n) \quad (5.3)$$

Equivalentemente vale dizer que é a data de término da última operação processada pelo sistema como um todo. Ao se minimizar este fator, implica em uma alta taxa de utilização das máquinas.

Violação da Pontualidade/Lateness/ L_i : Reflete o objetivo de terminar uma operação, na média, o mais assertivo possível. Definindo assertividade como entrega na data prometida (d_i) e a data efetivamente entrega (C_i). É positivo quando se atrasa e negativo quando se antecipa. É definido conforme abaixo:

$$L_i = C_i - d_i \quad (5.4)$$

Graficamente, a figura 5.2 exibe seu comportamento linear contínuo, supondo data de referência como sendo dia 15.

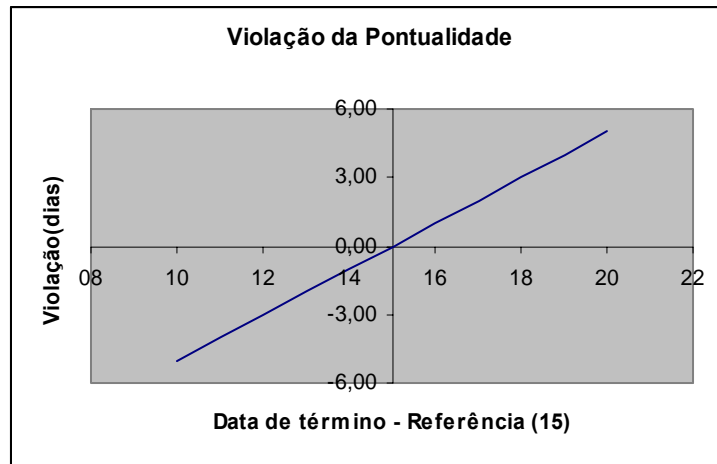


Figura 5.2 Ilustração da função violação

Máxima Violação/Maximum Lateness/ L_{\max} : O objetivo é uma função descontínua dada por

$$\max(L_1, L_2, \dots, L_n) \quad (5.5)$$

Implicando em tentar entregar a última tarefa o mais cedo possível.

Atraso/Tardiness/ T_i : Afirmar que um sistema atrasa o mínimo possível é o anseio de uma empresa. Este objetivo reflete o somatório dos atrasos de cada pedido. Também é uma função descontínua, e é definido por:

$$T_i = \max(C_i - d_i, 0) \quad (5.6)$$

A definição (5.6) denota que o atraso só é considerado no caso em que a data de término supera a data de entrega. Na situação em que a tarefa é antecipada, seu valor não é computado, independente de quanto tempo foi esta antecipação.

A figura 5.3 exibe seu comportamento, novamente tomando como base a referência 15.

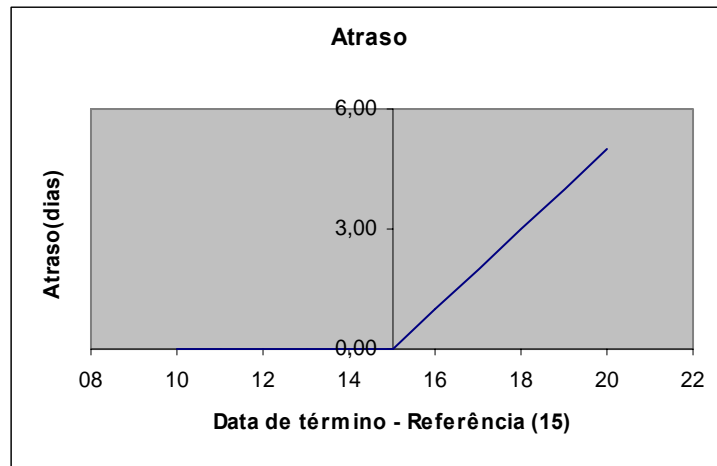


Figura 5.3 Ilustração da função atraso

Máximo Atraso/Maximum Tardiness/ T_{max} : Leva em consideração apenas o maior atraso. Isto implica em evitar postergar a entrega do pedido, porém conduz a possibilidade de inúmeros pedidos serem entregues fora do prazo. É definido por

$$\max(T_1, T_2, \dots, T_n) \quad (5.7)$$

Antecipação/Earliness/ E_i : Neste caso, deseja-se evitar antecipar operações. Este objetivo é significativo para empresas que possuam excesso de capacidade como por exemplo, com laticínios, pois antecipar em demasia a produção pode implicar em vencer a validade dos produtos. Esta função é o oposto T_i , leva-se em conta apenas a antecipação. Nesta função o que não é considerado é o atraso, isto é, definido em (5-9).

$$E_i = \max(d_i - C_i, 0) \quad (5.8)$$

Graficamente, a figura 5.4 exibe seu comportamento, para o mesmo caso da referência 15.

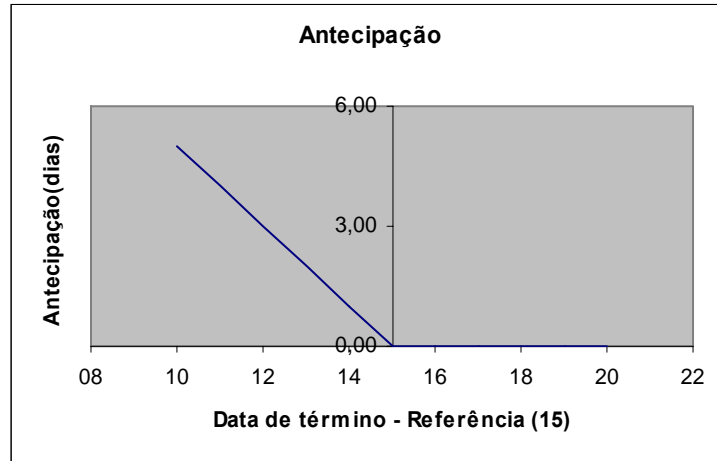


Figura 5.4 Ilustração da função antecipação

Máxima Antecipação/Maximum Earliness/ E_{max} : Esta função objetivo tenta evitar o término antecipado de uma operação. É definido por

$$\max(E_1, E_2, \dots, E_n) \quad (5.9)$$

Just In Time (JIT)/Earliness Tardiness/ ET_i : Atualmente é uma das funções objetivo mais desejadas nas empresas. Reflete o anseio de evitar entregar uma operação antes ou depois do prometido. É definida por

$$ET_i = E_i + T_i \quad (5.10)$$

Isto é, se uma tarefa é entregue antes do prometido tem-se $E_i > 0$ e $T_i = 0$; após a data ocorre o inverso $E_i = 0$ e $T_i > 0$. Para o caso da entrega no instante prometido, têm-se ambos $E_i = 0$ e $T_i = 0$. Tem-se a ilustração gráfica desta função na figura 5.5:

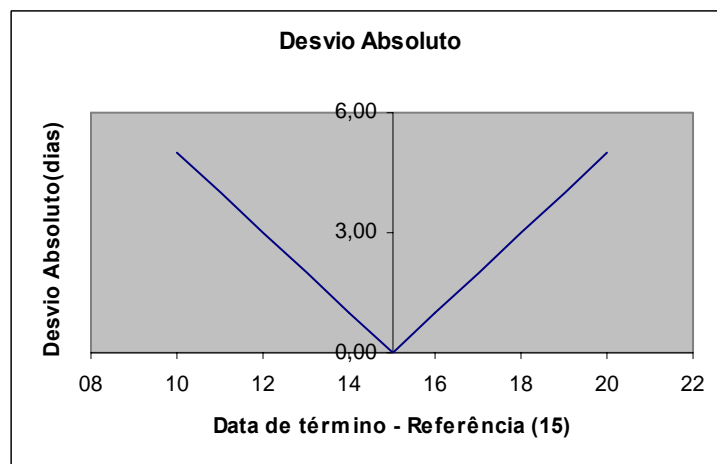


Figura 5.5 Ilustração da função Just in Time

Número de Pedidos Atrasados/ U_i : Considera apenas se um pedido está atrasado ou não sendo definida por:

$$U = \begin{cases} 1, & \text{se operação atrasada} \\ 0, & \text{qualquer outro caso} \end{cases} \quad (5.11)$$

Esta função leva em conta somente a quantidade de operações atrasadas, mas não verifica o quanto se está atrasado e não evita antecipação demasiada.

Da mesma forma, as funções acima podem ser generalizadas, acrescentando-se um peso ou prioridade w_i . Uma possível enumeração é exibida na tabela 5.1 a seguir.

Tabela 5.1 Resumo das funções objetivos derivadas

$\sum f(w_i c_i)$	$f_{\max}(w_i c_i)$	$\sum f(c_i)$	$f_{\max}(c_i)$
wC	wC _{max}	C	C _{max}
wL	wL _{max}	L	L _{max}
wE	wE _{max}	E	E _{max}
wT	wT _{max}	T	T _{max}
wEwT		ET	
wU		U	

Por exemplo, assumindo que w é o peso a ser considerado para uma dada tarefa, obtém-se, partindo-se, por exemplo, do critério atraso (T) pode-se derivar a minimização dos atrasos multiplicados pelo peso wT , a minimização do maior atraso multiplicado pelo peso wT_{\max} , a minimização da somatória dos atrasos T , ou a minimização do maior atraso T_{\max} .

Algumas outras funções objetivo destacadas são: “data de entrega penalizada e amortizada”, referenciada em (Pinedo, 2002), e “tempo de espera” (W), definida em (Jain, 1988), conforme pode ser vistos nas definições (5.12) e (5.13).

$$Amortizada = w_i \cdot (1 - e^{-r c_i}) \quad (5.12)$$

$$W_i = C_i - r_i - \sum p_j \quad (5.13)$$

Para encerrar é preciso destacar um objetivo genérico dado por:

$$w_{a_i} E_i + w_{b_i} T_i \quad (5.14)$$

O objetivo é contemplar diferenças entre antecipar e atrasar. Isto porque ao atrasar, os custos envolvidos, geralmente, são superiores aos custos de antecipar, Envolvendo multas, desistências, entre outros. Para o caso de $w_a = w_b$, recai-se à função wET e é possível observar que wE , E , wT e T são derivadas deste objetivo para o caso de $w = 0$ ou $w = 1$.

5.5 Representação

Há duas maneiras de se representar um sequenciamento: Gráficos de Gantt e Grafos Disjuntivos, conforme (Jain, 1988).

5.5.1 Gráfico de Gantt

A representação mais usada é o gráfico de Gantt. Este gráfico é uma forma matricial que ilustra a relação entre tarefas e máquinas, exibindo as datas de início, término e ociosidades. Assume-se que o tempo é um múltiplo de alguma unidade temporal como horas, dias ou semanas.

Nesta representação o eixo x representa o tempo e o eixo y representa os recursos. Barras com a duração de tempo necessário para processar a operação, p_i , são inseridas paralelas ao eixo x , na ordenada correspondente ao recurso que a processa. A figura 5.6 exibe um exemplo.

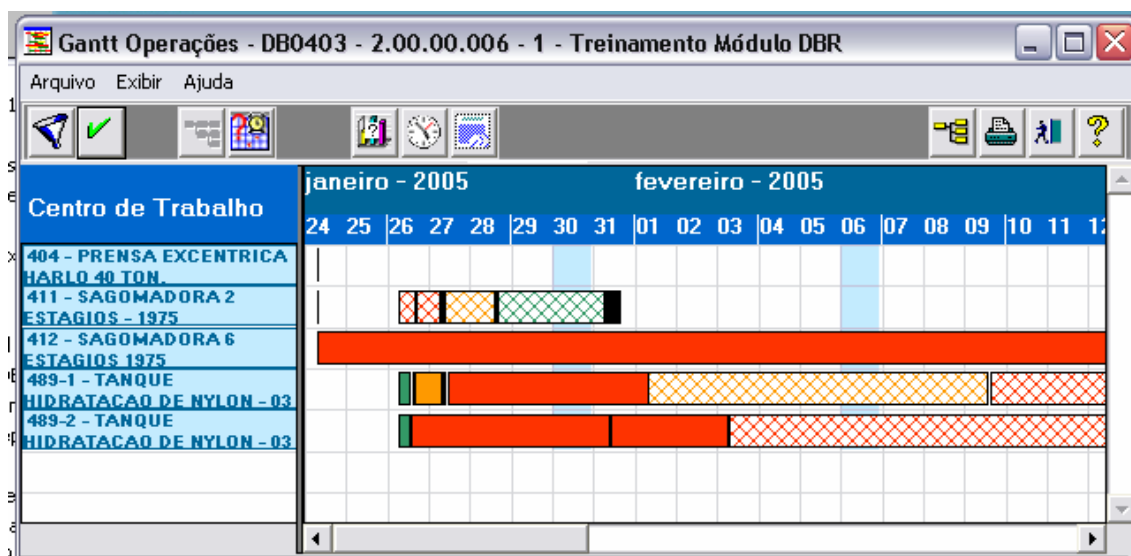


Figura 5.6 Exemplo de um gráfico de Gantt

Informações adicionais podem ser inseridas, como na figura 5.6. As diferentes cores ilustram a situação de uma operação (em dia, na iminência de atrasar e atrasada), bem como seu status ou situação (firme, isto é, já em processamento ou atendendo a um pedido e planejada) diferenciando-se pelo hachurado. Em um modelo mais completo, insere-se informações de tarefa e operações. Para o caso da figura acima, uma área de informações aparece ao se selecionar uma dada tarefa.

5.5.2 Grafos Disjuntivos

Outra forma interessante de se exibir um problema de sequenciamento é através de um modelo de grafo disjuntivo (Jain, 1988). Um grafo é uma tripla $G = \{N, A, E\}$ composta de nós (N), arestas (A) e pesos (E). Para o caso do sequenciamento é utilizado um grafo com nós ponderados onde N representa o conjunto de tarefas, A representa o conjunto de restrições e o conjunto de máquinas, as alternativas e E é o tempo de processamento das operações. Cada vértice representa uma operação a ser processada em um conjunto de máquinas, as alternativas. Adicionam-se os nós \odot e \star que correspondem a uma operação inicial fictícia e outra final fictícia. Os pesos de cada nó equivalem ao tempo de processamento de uma operação e assume-se que $p_{\odot} = p_{\star} = 0$. As datas de início e fim destes nós representam as datas de início e fim do problema de sequenciamento Π_j . \odot é conectado com a operação inicial de cada tarefa e similarmente as operações finais conectadas a \star .

A é um conjunto de arcos conjuntivos e disjuntivos. Os arcos conjuntivos são unidirecionais e representam as restrições de precedência, tal que, $(i, j) \in A$ indica que a operação i é uma predecessora imediata de j ($i \prec j$) na cadeia de operações da tarefa. As restrições disjuntivas são representadas por arcos bidirecionais. A cada membro de A é associado um par de arcos disjuntivos, que requerem uma mesma máquina, tal que $[i, j] = \{(i \prec j) \text{ ou } (j \prec i)\}$ e $\{i, j \in N\}$. Assim o peso do arco disjuntivo $(i, j) \in A$ é p_i ou p_j . Seu valor é p_i se $i \prec j$ ou p_j caso contrário.

As figuras abaixo representam a instância de um problema com 3 recursos, dadas por $\mathcal{N}_1 = \{O_1, O_3, O_6, O_{12}\}$, $\mathcal{N}_2 = \{O_4, O_5, O_{10}, O_{11}\}$ e $\mathcal{N}_3 = \{O_2, O_7, O_8, O_9\}$ e 4 tarefas com 3 operações cada, a saber $T_1 = \{O_1, O_5, O_9\}$, $T_2 = \{O_2, O_6, O_{10}\}$, $T_3 = \{O_3, O_7, O_{11}\}$ e $T_4 = \{O_4, O_8, O_{12}\}$. A figura 5.7 exibe o problema e a figura 5.8 a solução ótima para o problema, considerando como função objetivo minimização do *Makespan*.

A primeira exibe através dos arcos conjuntivos as restrições de precedência $\{\odot, O_1, O_5, O_9, \star\}$, $\{\odot, O_2, O_6, O_{10}, \star\}$, $\{\odot, O_3, O_7, O_{11}, \star\}$ e $\{\odot, O_4, O_8, O_{12}, \star\}$. Os demais arcos representam os disjuntivos. Pode-se subdividi-los em subconjuntos, um para cada máquina. Por exemplo, seja o conjunto $O = \{O_1, O_3, O_6, O_{12}\}$ das operações que são processadas na máquina 1. Para cada operação O_i pertencente ao conjunto O , existe um arco disjuntivo bidirecional com peso $p_k = p_i$ ou p_j que liga O_i aos demais O_j pertencentes a O . Para o caso do conjunto O os arcos disjuntivos seriam formados pelos pares ordenados $\{(O_1, O_3), (O_1, O_6), (O_1, O_{12}), (O_3, O_1), (O_3, O_6), (O_3, O_{12}), (O_6, O_1), (O_6, O_3),$

$(O_6, O_{12})\}$. A disjunção é dada pela proposição lógica $(O_1, O_3) = \text{Verdadeiro} \Leftrightarrow (O_3, O_1) = \text{Falso}$ ou seja, se a operação 1 antecede 3, a operação 3 **não** antecede 1.

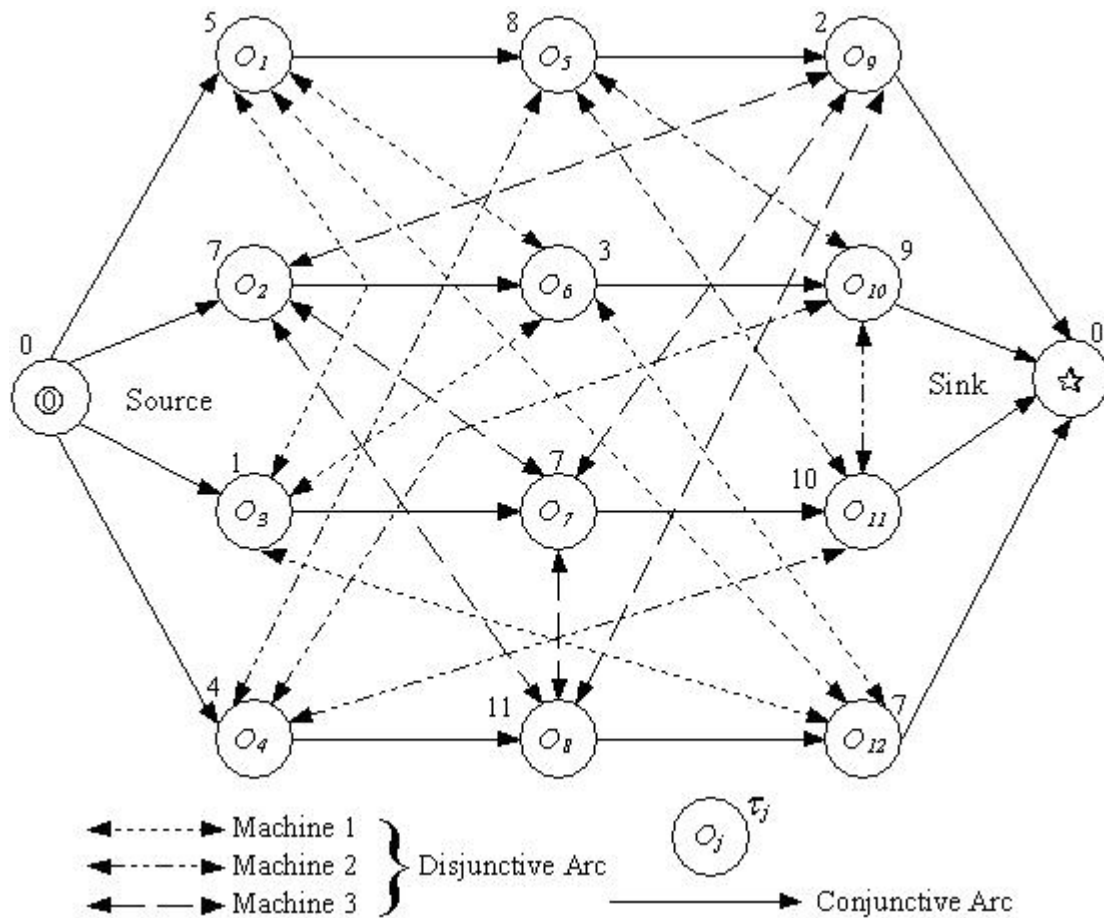


Figura 5.7 Representação do problema citado no texto, extraído de (Jain, 1988)

A interpretação da figura 5.8 é a seguinte: na máquina 1 a sequência executada é $\{O_1, O_3, O_6, O_{12}\}$, na máquina 2 a ordem é $\{O_4, O_5, O_{10}, O_{11}\}$ e por fim na máquina 3 a ordem é dada por $\{O_2, O_7, O_8, O_9\}$. Os valores juntos aos nós representam o tempo de processamento da operação correspondente. Posicionando as operações no tempo e respeitando as restrições de precedência obtém-se para a tarefa 1 as seguintes datas de início de suas operações (0, 7, 25) com data de término de 27. Para a tarefa 2 os valores são (0, 7, 13) com data de término 22. Para a tarefa 3 obtém-se (5, 7, 22) com data de término 32 e por fim a tarefa 4, com (0, 14, 25) com data de término 32. O valor da função objetivo é então dado por $27 + 22 + 32 + 32$, ou seja, 113. Sendo este o valor mínimo para a instância Π_j em questão.

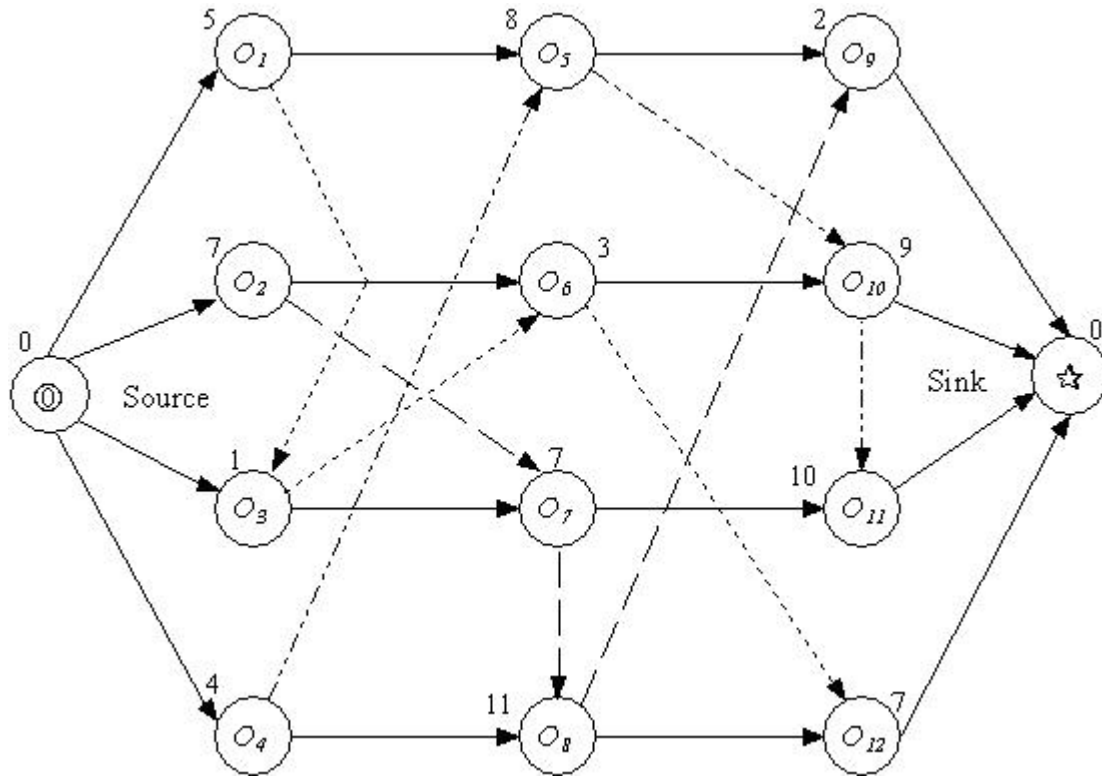


Figura 5.8 Solução ótima para o problema da figura 5.7, extraído de (Jain, 1988)

5.6 Complexidade

5.6.1 Noções Gerais

Após os trabalhos de Cook, referências (Cook, 1971), (Cook, 2000) e (Cook, 2002), e Levin, referência (Levin, 1973), surgiu uma nova questão matemática. Esta questão faz parte dos desafios deste milênio e é conhecida como o problema “ P versus NP ”. A seguir uma breve descrição da conjectura, baseada em (Myiazawa, 2001).

Seja I uma instância de um dado problema combinatorial, por exemplo, circuito hamiltoniano. Seja A um algoritmo aplicado sobre I . Seja ainda A_d e A_c dois algoritmos distintos, de forma que o primeiro retorne como resposta a informação se existe ou não solução, ou se a solução é ótima ou não e o segundo retorne a solução completa do problema da instância I .

A questão é, será que é possível escrever algoritmos A_c em função de A_d ? Se sim, a pergunta mais geral é será que toda instância de I , permite isto? Por fim, será que qualquer problema parecido com I admite este tipo de solução.

Para estes problemas combinatoriais, pode-se classificá-los em problemas do tipo P ou do tipo NP . Problemas P são problemas resolvíveis em tempo polinomial e

consistem em aplicar algoritmos do tipo A_d , enquanto que problemas NP são problemas do tipo A_c , os quais não se conhecem um algoritmo que o resolva em tempo polinomial. Desta forma, um problema NP , pode ser transformado em um problema P , bastando para tal encontrar um algoritmo que o resolva em tempo polinomial.

A transformação de NP em P é dependente da existência de um algoritmo que o resolva em tempo polinomial. Esta dependência será que é temporal ou impossível? Se for possível provar que todo problema NP pode ser transformado em P , então sua solução é uma questão de tempo e assim $P = NP$, porém se $P \neq NP$ for provado, não existe algoritmos em tempo polinomial que resolva NP e a busca se transforma em encontrar um algoritmo estocástico, que no caso médio, convirja para a solução ótima (Levin, 1973).

As implicações, segundo Cook, de provar que $P = NP$ trariam para indústria a possibilidade de se resolver todos os problemas combinatoriais em tempo polinomial. Ao mesmo tempo, teriam resultados catastróficos para a segurança computacional, provando que os algoritmos criptográficos, baseados em fatoração de números primos, poderiam ser quebrados em tempo computacional reduzido.

Desde o surgimento da conjectura, (Cook, 1971), muitos pesquisadores, engenheiros, estudiosos, entre outros têm tentado sem sucesso o desenvolvimento de tais algoritmos. Isto provocou o surgimento da conjectura que $P \neq NP$. O que ainda não foi provado, e é justamente um dos prêmios do milênio, (Devlin, 2004).

Atualmente, a configuração (existem muitas outras) dos problemas se enquadra em P , NP , NP -Completo e NP -Difícil. A figura 5.9, extraída de (Miyazawa, 2001), mostra esta configuração. Um problema é dito NP -completo se este pode ser reduzido a um problema P . Portanto, resolver a instância em P , resolve a instância em NP . De outro lado, um problema é dito NP -difícil se este pode ser reduzido a P em tempo polinomial. Perceber a diferença que NP -completo, reduz-se a P que é resolvido em tempo polinomial e NP -completo reduz-se a P em tempo polinomial, mas ainda é necessário resolvê-lo.

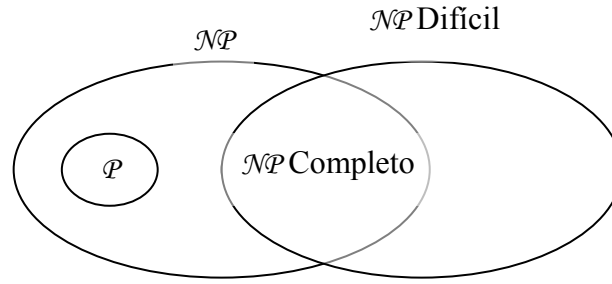


Figura 5.9 Possível configuração para as classes \mathcal{P} , \mathcal{NP} e \mathcal{NP} Difícil

Para exemplificar esta distinção, baseando-se no exemplo de (Miyazawa, 2001), seja I uma instância de um problema de encontrar um circuito hamiltoniano em um grafo. Seja A_d um algoritmo que aplicado a I , retorne “SIM” para o caso de existir tal circuito, ou “NÃO” caso contrário. Seja A_c um algoritmo que encontra tal circuito. Se for encontrado o algoritmo A_d , é possível desenvolver um algoritmo A_c , que executa em tempo polinomial, da seguinte forma:

```

Algoritmo  $A_c$  (Instância  $I$ )
  Início
    Se  $A_d(I)$  = “SIM” Então
      Faça  $H_0 = I$ 
      Para  $i = 1$  até Número de Arestas Faça
        //  $e_i$  é uma aresta do Grafo  $I$ 
        Se  $A_d(H_{i-1} - e_i)$  = “SIM” Então
           $H_i = H_{i-1} - e_i$ 
        Senão
           $H_i = H_{i-1}$ 
        FimSe
      FimPara
    Senão
      “Não há circuito Hamiltoniano!”
    FimSe
  FimAlgoritmo

```

Algoritmo 6.1 – Transformação de NP em P

Para os problemas de sequenciamento, tendo em vista as observações, se for provado que um problema é do tipo NP -difícil, é provável que não exista um algoritmo que o resolva em tempo polinomial. Do contrário, é possível desenvolver um procedimento que o resolva em tempo computacional aceitável. A seção, a seguir, mostra como relacionar os problemas de sequenciamento, reduzindo-os a problemas mais simples.

5.6.2 Redução dos problemas de sequenciamento

Baseado nos trabalhos de (Utku, 1999), reproduz-se aqui a árvore de reducibilidade dos problemas de sequenciamento conforme discutido na seção anterior. A figura 5.10 exibe uma árvore de redução.

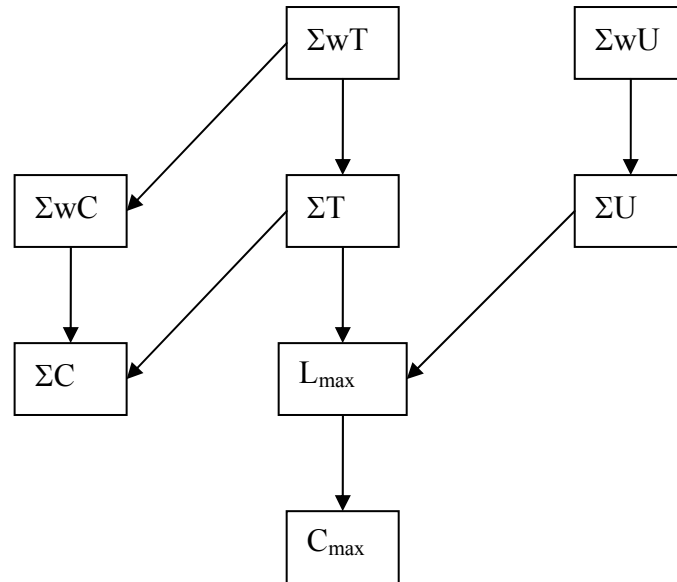


Figura 5.10 Árvore de redução dos problemas de sequenciamento

Ela informa, por exemplo, que o problema L_{max} pode ser reduzido ao problema de C_{max} . Isto é representado pela simbologia $L_{max} \succ C_{max}$. Quanto mais acima na árvore, mais difícil a sua solução. Isto implica em dizer que o problema mais difícil de resolver é $w_i T_i$.

6 Heurística Proposta

6.1 Introdução

As situações vivenciadas no chão de fábrica fogem do escopo do estudo teórico. Os modelos criados têm por características absorver a essência dos problemas de sequenciamento. A definição clara dos objetivos nem sempre é possível, o que recai em múltiplos objetivos ou objetivos conflitantes. A não adequação da teoria tem origem muitas vezes devido às técnicas ou maneiras particulares que cada um adota - as heurísticas. Isto ocorre devido a imposições de mercado, máquinas, equipamentos ou tecnologia. Tentando unir as exigências mercadológicas e a teoria, a heurística proposta possui um híbrido entre os modelos acadêmicos e as necessidades práticas.

O ambiente fabril possui inúmeros centros de trabalho ou máquinas os quais, por sua vez, processam inúmeros itens. Esta característica é especialmente verdadeira para ambientes *job shops*, ou seja, ambientes em que cada produto consiste de um conjunto de operações, que seguem por roteiros distintos e passam por máquinas ou grupo de máquinas específicas.

Esta diversidade de produtos produzidos implica em inúmeras tarefas que devem ser organizadas para que o objetivo principal da empresa seja atingido. Por consequência, e conforme visto na seção 5.6, o número de variáveis e possibilidades de solução é muito grande.

Então, para reduzir o domínio do problema, ampara-se na teoria das restrições, com o objetivo de diminuir o número de variáveis decisórias e justificar a aplicação da heurística proposta apenas aos centros de trabalho restritivo, ou seja, o recurso com a maior razão demanda dividida pela capacidade, conforme visto na seção 4.3.

As modificações efetuadas na heurística visam atender alguns requisitos de mercado tais como, trabalhar com capacidades diferentes, centros de trabalho válidos e ferramentais. Para isto, utilizaram-se técnicas de programação dinâmica multirotulada (Daru, 2005) para efetuar o balanceamento das cargas tanto das ferramentas quanto dos centros de trabalho. O objetivo definido foi a minimização do atraso das tarefas prioritárias, e para isto utilizou-se uma regra de despacho dinâmica conhecida como *Custo Aparente do Atraso (ATC – Apparent Tardiness Cost)*, adaptando-se o algoritmo original proposto por (Rachamadugu, 1982) para contemplar questões de datas mínimas.

Para compreender os limites da heurística, detalha-se a seguir as hipóteses acerca do modelo.

6.2 Hipóteses e Constatações

Para formalizar e situar o problema dentro da teoria do sequenciamento, a heurística é classificada, usando-se a notação $\alpha|\beta|\gamma$, como ***Rm/ rj Mj Fj brkdw / wT***. As hipóteses assumidas aqui são destacadas abaixo:

Rm Ambiente com máquinas não relacionadas entre si, mas que processam uma mesma operação. A velocidade de processamento de uma operação em um determinado recurso é dada por v_{ij} .

rj Cada operação possui uma data mínima ou data mais cedo que esta pode iniciar. Antes desta data o sistema impede sua execução.

Mj Apesar do grupo de máquinas possuir centros de trabalhos similares, estes podem, por questões tecnológicas, não permitir o processamento de uma dada operação. A lista de centros permissíveis é dada pelo vetor *Mj*.

Fj Não obrigatoriamente, mas possível, uma operação exigir uma ferramenta específica. Isto implica que outra operação que a use, só possa ser iniciada após a conclusão da operação antecessora em relação à ferramenta e não ao centro de trabalho, sendo um fator operacional complicante para resolução do problema.

brkdw Os calendários de trabalho podem ser distintos para cada recurso. Isto implica em um tratamento especial aos períodos de cálculo, devido a paradas para manutenção e turnos de trabalho diferentes.

wT Esta é a função objetivo definida, minimizar o atraso das operações prioritárias.

Outras hipóteses são:

- ☒ O sistema não distingue tempos de preparação, ou também chamado de tempo de *setup*, entre operações. Sendo este fixo e incorporado na variável p_{ij} .
- ☒ O sistema não permite recirculação, isto é, tarefas que possuem mais de uma operação que passem pela restrição, e que não estejam em estrutura paralela.
- ☒ O sistema não leva em conta a capacidade dos centros de trabalho não restritivos.

- ☑ O sistema não distingue períodos de parada⁵ para os centros, apenas diminui a capacidade total do recurso; para o sistema é indiferente se o período de parada é no início do planejamento ou no final.

6.3 Localização da Função Sequenciamento

Para compreender onde se insere o sequenciador propriamente dito, faz-se necessário contextualizá-lo dentro do planejamento operacional. A figura 6.1 exhibe as fases de um sistema de planejamento fino, baseado na teoria das restrições. Em destaque a localização da função sequenciamento. A figura 6.2 exhibe a interface de um sistema comercial, o qual ilustra a aplicação prática destas funções. Cada fase é descrita sucintamente a seguir:

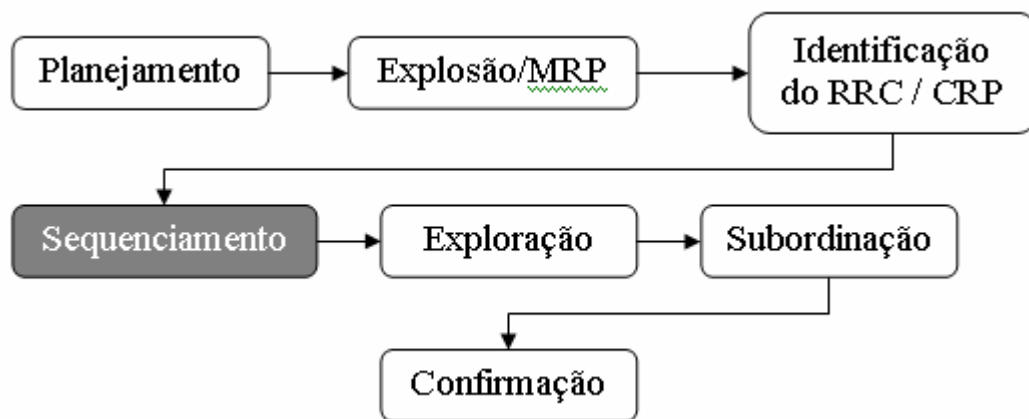


Figura 6.1 Fases de um sistema de planejamento fino, baseado na teoria das restrições

⁵ Período de parada é definido como um período o qual o recurso, no caso um centro de trabalho, não está disponível. Por exemplo, uma manutenção.

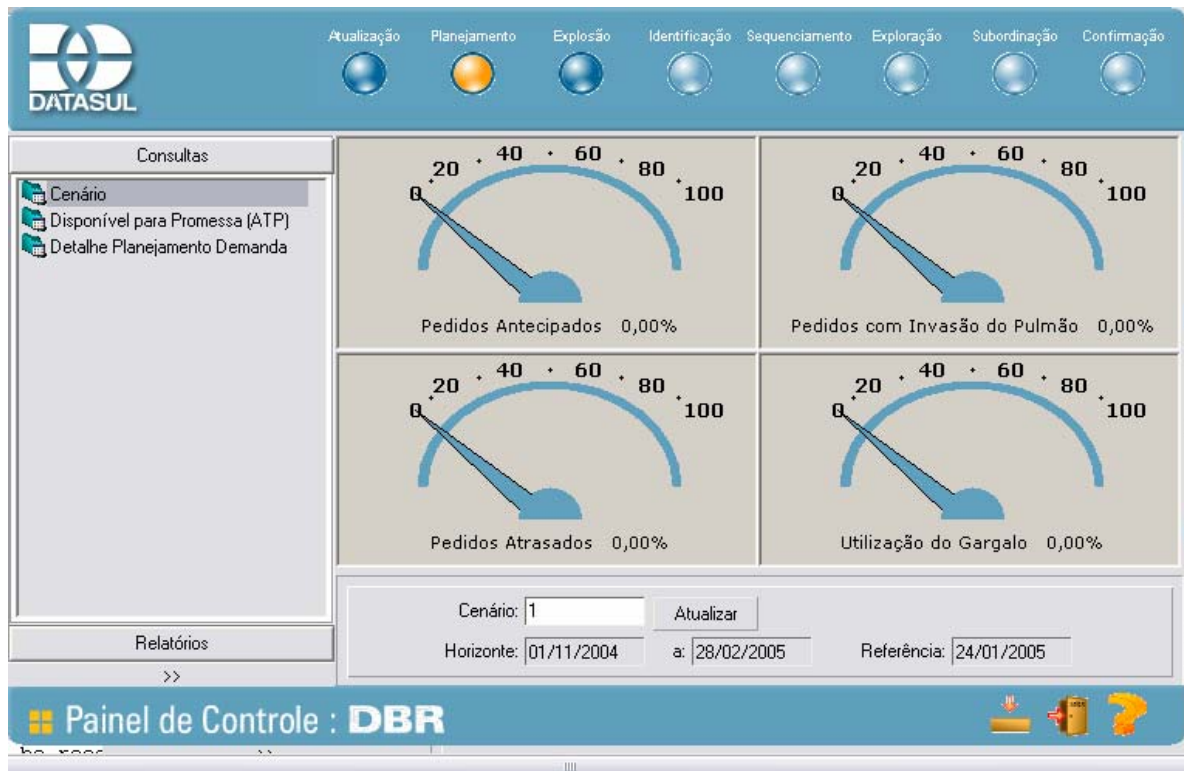


Figura 6.2 Interface de um sistema comercial de planejamento fino e suas fases

Planejamento Esta fase consiste em verificar quais são as necessidades reais de produção. Verifica-se o que se tem de pedido e previsões, confronta-se com o que se tem em estoque e produção (conhecidas como ordens firmes). O resultado deste cálculo é a quantidade a produzir de itens acabados. De posse desta informação, aplica-se a política do cliente para determinar a quantidade final. Entenda-se por política, técnicas para minimizar custos produtivos como, por exemplo, lote econômico, lote múltiplo, lote mínimo, nível superior, make to order, entre outros. Estes conceitos podem ser vistos no capítulo 3 deste trabalho. A figura 6.3 exibe um resumo do cálculo realizado para um determinado item.

	Atraso	Futuro
Previsão de Vendas(Bruta)	0,0000	0,0000
Previsão de Vendas(Líquida)	0,0000	0,0000
Pedidos de Venda	50,0000	0,0000
Demanda Total	50,0000	0,0000
Ordens Firmes	800,0000	0,0000
Ordens Planejadas	0,0000	0,0000
Estoque Projetado	1.660,00...	1.660,00...
ATP	0,0000	1.660,00...
ATP Acumulado	0,0000	1.660,00...

Figura 6.3 Informações disponíveis após os cálculos da fase de planejamento

Explosão Uma vez determinado “o que” deve ser produzido e “para quando”, deve-se identificar “o que é necessário” e “quais recursos” devem ser utilizados. Esta fase é similar ao cálculo realizado pelo *MRP*, com o adendo de levar em consideração os calendários de trabalho e calcular as datas mais cedo e mais tarde (data mínima e data de entrega) para cada material e operação. Estes cálculos são exibidos na figura 6.4, onde é possível visualizar o tipo da ordem (firme - *OF* ou planejada - *OP*), e informações de quantidades e datas. Para cada uma destas ordens é realizada uma explosão em sua estrutura para identificar as quantidades necessárias e calcular os tempos de processamento necessários. É a fase que realiza todos os cálculos necessários de entrada do sequenciador. A figura 6.5 exhibe a estrutura de uma ordem específica, bem como os materiais e operações necessários para compor o item. As datas mais cedo e mais tarde estão exibidas ao lado.

Datsul DBR					
Consulta Ordens Produção					
Cenário		1			
Descrição		teste			
Horiz Cálculo		01/11/04-28/02/05			
Tipo	Ordem	Item	Quantidade	Data de Entrega	Prioridade
OF	5.508	07.01.0370	459,00	26/07/2004 17:00	99
OF	51.928	02.03.0200	50,00	06/08/2004 17:00	99
OF	5.657	07.01.0055	470,00	30/09/2004 17:00	99
OF	5.690	07.01.1010	8,00	30/09/2004 17:00	99
OF	55.129	03.54.0220	55,50	30/09/2004 17:00	99
OF	5.696	07.01.0890	94,00	29/10/2004 17:00	99
OF	5.706	07.01.1040	130,00	10/11/2004 17:00	99
OF	55.687	03.58.0050-025	3,00	11/11/2004 17:00	99
OF	5.728	07.01.0070	606,00	12/11/2004 17:00	99
OF	5.754	07.99.0050	168,00	18/11/2004 17:00	99
OF	55.708	02.08.0070	800,00	18/11/2004 17:00	99
OF	55.193	01.01.0190	300,00	22/11/2004 17:00	99
OF	55.203	01.01.0261	200,00	22/11/2004 17:00	99
OF	5.755	07.99.0060	75,00	25/11/2004 17:00	99
OF	5.714	07.01.0255	5.250,00	30/11/2004 17:00	99
OF	5.760	07.01.0080	288,00	30/11/2004 17:00	99
OF	5.761	07.01.0070	1.784,00	30/11/2004 17:00	99
OF	56.350	07.01.0285	1.888,00	30/11/2004 17:00	99
OF	5.777	07.01.0325	1.182,00	20/12/2004 17:00	99
OF	56.152	01.01.0140	2.000,00	28/12/2004 17:00	99
OF	56.154	01.01.0141	500,00	28/12/2004 17:00	99
OF	56.156	01.01.0143	500,00	28/12/2004 17:00	99

Figura 6.4 Ordens geradas após a fase de explosão

Consulta Rede Operações - DB0402 - 2.00.00.006 - 1 - Treinamento Módulo DBR							
Arquivo Ajuda							
<div> <div>Ordem: 121</div> <div>Entrega: 21/01/2005 17:00</div> <div>Reprogramada: 08/02/2005 16:00</div> </div> <div> <div>Item: 05.01.0070 - A027-BRAC M752</div> <div>Quantidade: 3.000,0000 PC</div> <div>Pedido: 0 Cliente: Pulmão Exp: 9</div> </div>							
	Item	Quantidade ...	Quantidade ...	Quantidade ...	Data Início mais Cedo	Data Início/Disponibil...	Data Término Inicial
Ordem Produção Planejada:	121 05.01.0070	3.000,...	---	---	07/02/2005 16:00	08/02/2005 16:00	21/01/2005 17:00
Fabricado	05.01.0070	3.000,0000	3.000,0000	3.000,0000	07/02/2005 16:00	07/02/2005 16:00	---
Operação TSP: 10		3.000,0000	---	---	01/02/2005 07:00	?	20/01/2005 17:00
Fabricado	05.01.0070 NAT	3.000,0000	3.000,0000	3.000,0000	01/02/2005 07:00	?	---
Operação: 10		3.000,0000	---	---	01/02/2005 07:00	?	14/01/2005 08:00
Comprado Restritivo	06.47.0070	0,0000	0,0000	42,0000	01/02/2005 07:00	?	---

Figura 6.5 Cálculos realizados durante a explosão para a rede de operações/lista de material gerada

Identificação do RRC Uma vez identificado tudo o que é necessário produzir, e sabendo quais recursos são necessários para realizar uma operação e seus tempos, calcula-se a carga necessária para atender a demanda, dentro do horizonte especificado. A figura 6.6 exibe um gráfico carga máquina, no qual se pode identificar o recurso com

restrição de capacidade. Após definido qual é o recurso, que geralmente é um grupo de máquina (um conjunto de recursos), passa-se para a próxima fase.

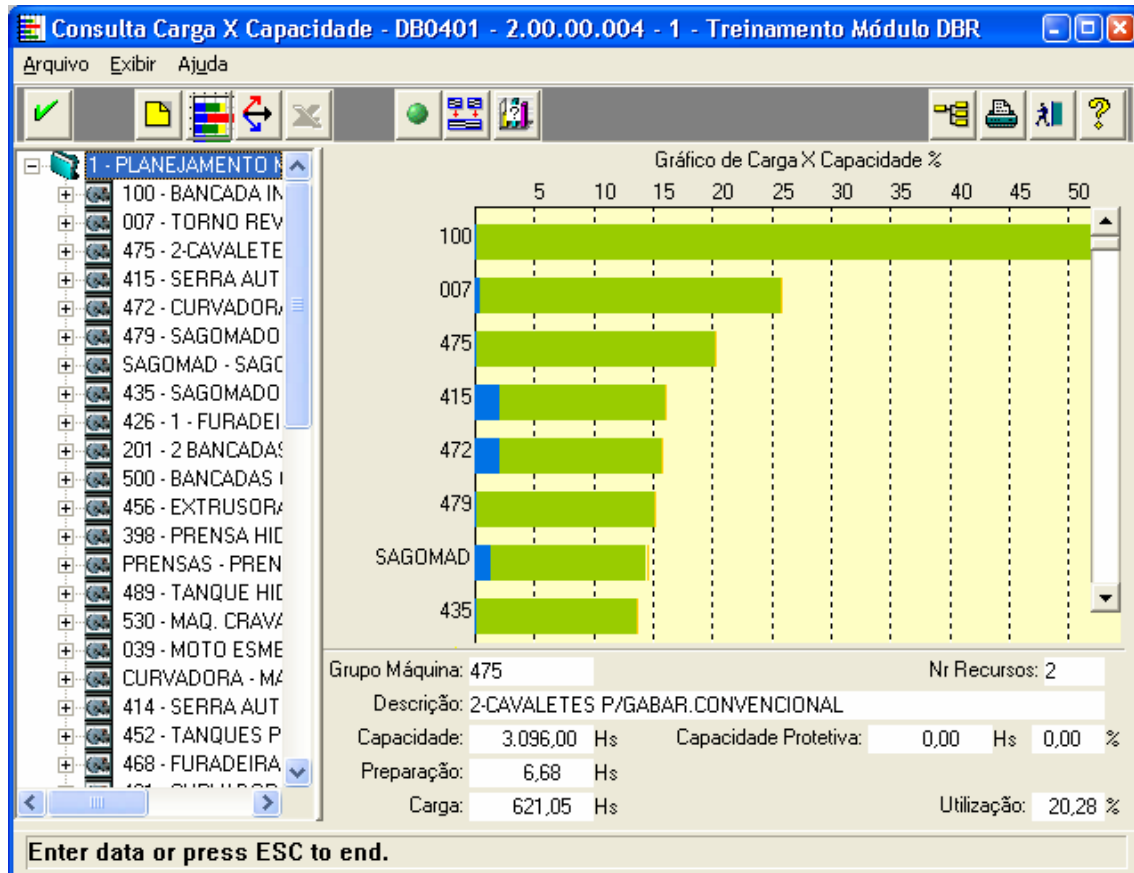


Figura 6.6 Carga máquina para identificação do recurso com restrição de capacidade

Sequenciamento Para um resumo do sequenciamento, a figura 6.7 exibe as suas fases, que serão descritas em detalhes na seção 6.4.

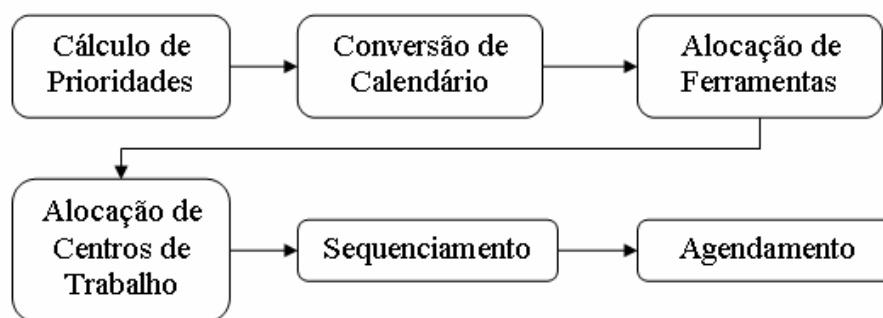


Figura 6.7 Fases do sequenciamento

Exploração Inúmeras empresas possuem trabalho durante 24hs. Para estas, existe a possibilidade de reduzir *setups* ou descarregar operações para terceiros. As que não o fazem, tem a possibilidade de empregar horas extras. Para isto é que serve a exploração, ou seja, aumentar a capacidade dos recursos restritivos. Nesta fase, é

aumentado o período de trabalho dos recursos. Isto acarreta um aumento de disponibilidade que altera a programação. Após esta fase é realizado o sequenciamento novamente para refletir as alterações nas programações. A figura 6.8 exibe um exemplo de interface de exploração.

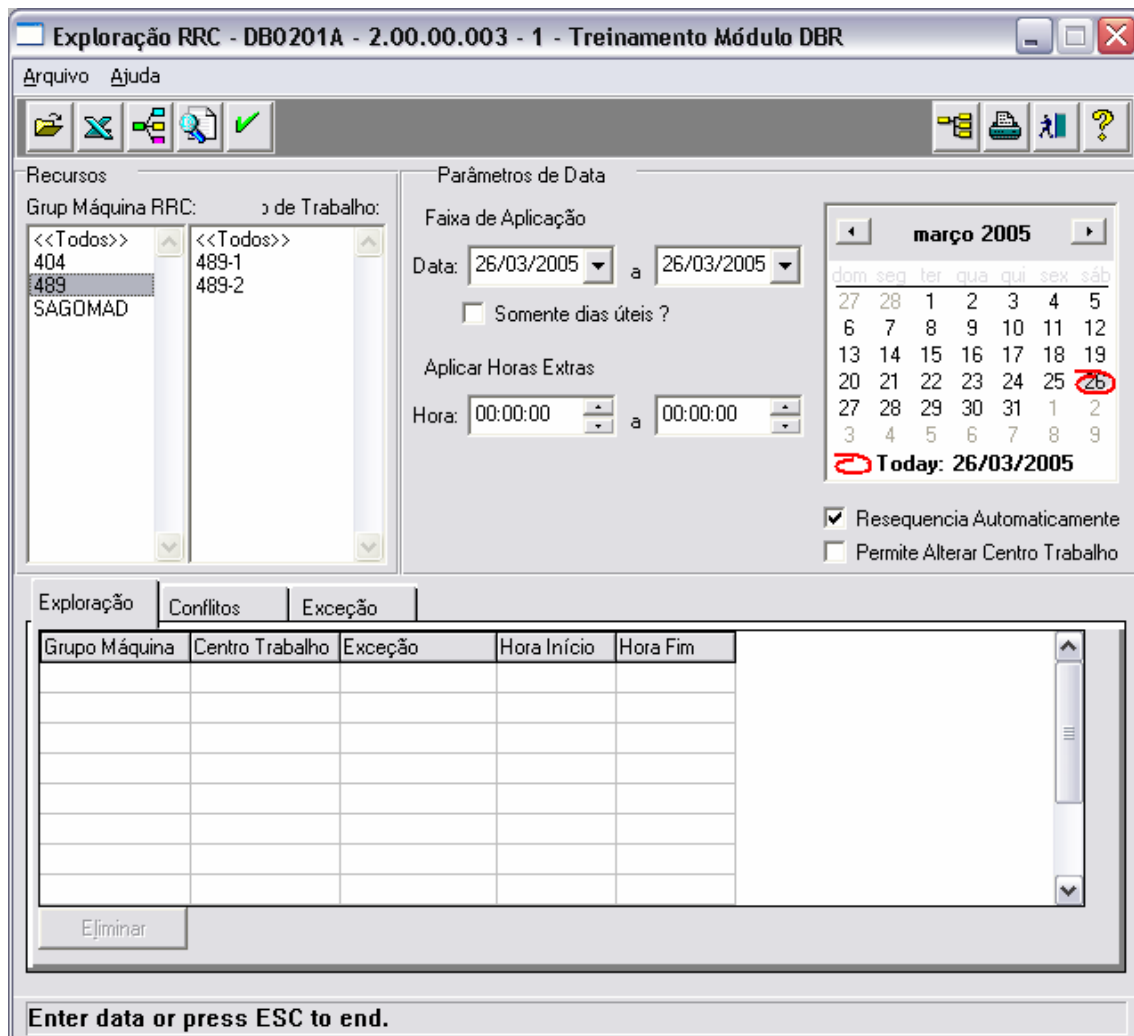


Figura 6.8 Interface de exploracao para aplicação de horas extras

Subordinação Levando-se em conta que o ritmo da produção como um todo é determinado pela restrição, os demais recursos produtivos são apenas notificados de quando o material a ser processado no gargalo deve estar pronto. Esta fase consiste em determinar as datas das demais operações da estrutura de cada material. Não é levado em consideração se um determinado recurso possui ou não capacidade para a data imposta, isto porque, se for levado em consideração que o recurso mais carregado é o que se está sequenciando, os demais possuem capacidade superior e, portanto podem realizar as operações determinadas sem problemas. A figura 6.9 exibe os cálculos das datas das operações não restritivas.

Item	Quantidade Programada	Data Início mais Cedo	Data Início/Disponibilidade	Data Emissão	Data Término Inicial
Ordem Produção Firme: 57.3...	50.0000	05/04/2005 11:36	14/04/2005 10:36	---	07/02/2005 17:00
01.07.0405	50.0000	05/04/2005 11:36	05/04/2005 11:36	---	---
Operação: 40	50.0000	05/04/2005 11:36	05/04/2005 11:36	---	28/01/2005 08:00
Operação: 30	50.0000	05/04/2005 11:36	05/04/2005 11:36	---	28/01/2005 08:00
Operação: 20	50.0000	05/04/2005 11:36	05/04/2005 11:36	---	28/01/2005 08:00
Operação: 10	50.0000	05/04/2005 11:36	05/04/2005 11:36	---	28/01/2005 08:00
Fabricado 03.58.0045	256.5000	24/01/2005 07:00	05/04/2005 11:36	---	---
Operação: 20	256.5000	24/01/2005 07:00	26/01/2005 07:00	---	28/01/2005 08:00
Operação: 10	256.5000	24/01/2005 07:00	26/01/2005 07:00	---	28/01/2005 08:00
Comprado Restritivo 04.02.0030	0.0000	24/01/2005 07:00	26/01/2005 07:00	21/01/2005 07:00	---
Comprado Restritivo 04.06.0030	0.0000	24/01/2005 07:00	26/01/2005 07:00	23/12/2004 07:00	---
Fabricado 07.01.0035	100.0000	05/04/2005 11:36	05/04/2005 11:36	---	---
Operação TSP: 10	100.0000	03/03/2005 07:00	02/03/2005 17:00	---	28/01/2005 08:00
Comprado Restritivo 03.63.0050	0.0000	03/03/2005 07:00	02/03/2005 17:00	23/01/2005 17:00	---
Comprado Restritivo 07.01.0039	0.0000	24/01/2005 07:00	05/04/2005 11:36	03/03/2005 11:36	---
Fabricado 07.01.0815	64.0000	25/02/2005 16:00	05/04/2005 11:36	---	---
Operação: 10	64.0000	25/02/2005 16:00	05/04/2005 11:36	---	28/01/2005 08:00
Fabricado 07.01.0778	0.0000	24/01/2005 07:00	05/04/2005 11:36	---	---
Fabricado 08.01.0311	64.0000	25/02/2005 16:00	05/04/2005 11:36	---	---
Operação TSP: 10	64.0000	21/02/2005 07:00	29/03/2005 13:36	---	28/01/2005 08:00
Comprado Rest...	0.0000	21/02/2005 07:00	29/03/2005 13:36	24/02/2005 13:36	---

Figura 6.9 Subordinação de operações não restritivas à restrição

Esta consideração está amparada pela teoria das restrições. Isto não garante que devido às restrições de precedência e estoques em processo a execução deste plano seja factível. Porém, para evitar ou minimizar a possibilidade de deixar a restrição sem material para processar, utiliza-se outros conceitos da teoria, como pulmão. O pulmão é uma proteção de tempo que garante que os recursos não restritivos antecipem a chegada do material ao *RRC*.

Ainda existe o problema quando os recursos tem capacidades muito próximas, porém a *TOC* não se aplica a estes ambientes, sugerindo o desbalanceamento da produção para caracterizar uma restrição e assim permitir sua aplicação.

Confirmação Após efetuada toda a programação, a fase final consiste em validá-la em um sistema *ERP* para posterior execução pelo chão de fábrica. A confirmação transforma a programação de ordens em programa de produção, e as ordens que eram planejadas passam a fazer parte do horizonte fixo e denominadas de ordens firmes.

6.4 A Heurística de Sequenciamento

Devido a cada ambiente produtivo possuir características próprias, a heurística visa atender um conjunto mínimo similar à maioria. Por isto a heurística permite, em conjunto com a *TOC*, atender a empresas que i) possuam uma restrição bem caracterizada, ii) visem minimizar o atraso dos pedidos prioritários. O algoritmo **não** atende a ambientes que possuam *setup* e ambiente produtivo com recursos que possuam capacidades ou cargas muito similares.

Sua aplicação se dá a qualquer ambiente produtivo que possua as características supra-citadas. Para citar um exemplo disto, empresas de fundição que possuem um forno como restrição. Outro exemplo são empresas de injeção de polímeros, que possuem a máquina injetora como restrição, desde que não possuam *setup* significativo. Outros ambientes como *job shops* se caracterizados em dados momento os recursos restritivos, também permitem aplicação.

Para os ambientes produtivos os quais a heurística se aplica, ver seção 6.2 Hipóteses e Constatações.

A heurística consiste em um conjunto de algoritmos interconectados, com o objetivo de encontrar uma solução satisfatória para o problema de sequenciamento. Cada algoritmo possui uma funcionalidade e atende a uma parte das restrições do problema total.

Estes algoritmos possuem objetivos de calcular as prioridades, calcular as disponibilidades, alocar ferramentas, alocar centros de trabalho, sequenciar operações e por fim agendá-las. Estas fases podem ser visualizadas na figura 6.10. A seguir as subseções descrevem com detalhes cada fase da heurística proposta.

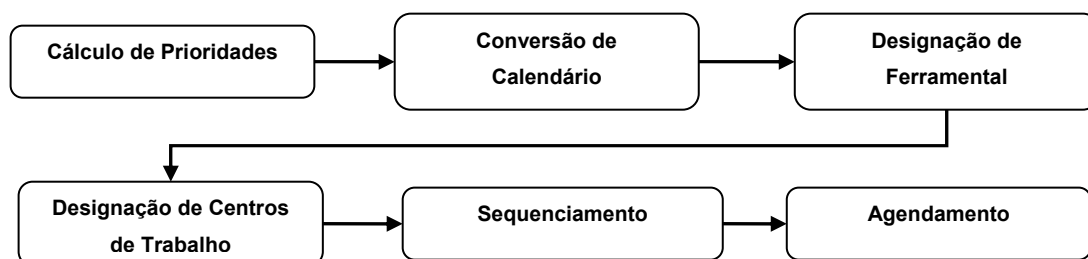


Figura 6.10 Fases do Sequenciamento

6.4.1 Cálculo de Prioridades

Em geral as empresas classificam seus produtos de acordo com algum critério, como por exemplo, vendas ou volume de produção, e de acordo com este critério os

priorizam. Estes são conhecidos como "carros chefes" e devem receber mais atenção, isto é, serem processados primeiro no ambiente produtivo. Uma forma simplificada de atender estes requisitos é permitir que o usuário informe para cada entidade (pedido, item, operação, etc) um valor correspondente a seu grau de importância. Por exemplo, em empresas exportadoras, divide-se os clientes em internacionais e nacionais. Um atraso em um pedido para um cliente internacional, devido ao tempo de transporte (data de embarque do navio), pode custar multas contratuais ou até perda do pedido/cliente. Claramente estes pedidos são mais prioritários e seu atraso é mais relevante do que um pedido para reposição de estoque. Estas necessidades é que justificam o uso de prioridades.

Para um entendimento do cálculo da prioridade, considere a figura 6.11 que representa a estrutura de um item, o qual possui uma operação restritiva (em destaque). Este item por sua vez possui uma prioridade que é retransmitida a operação. Ainda é necessário, porém, identificar a ordem a que este item atende. Esta ordem pode ser para atender pedidos (situação 2) ou para atender a outras ordens (situação 1).

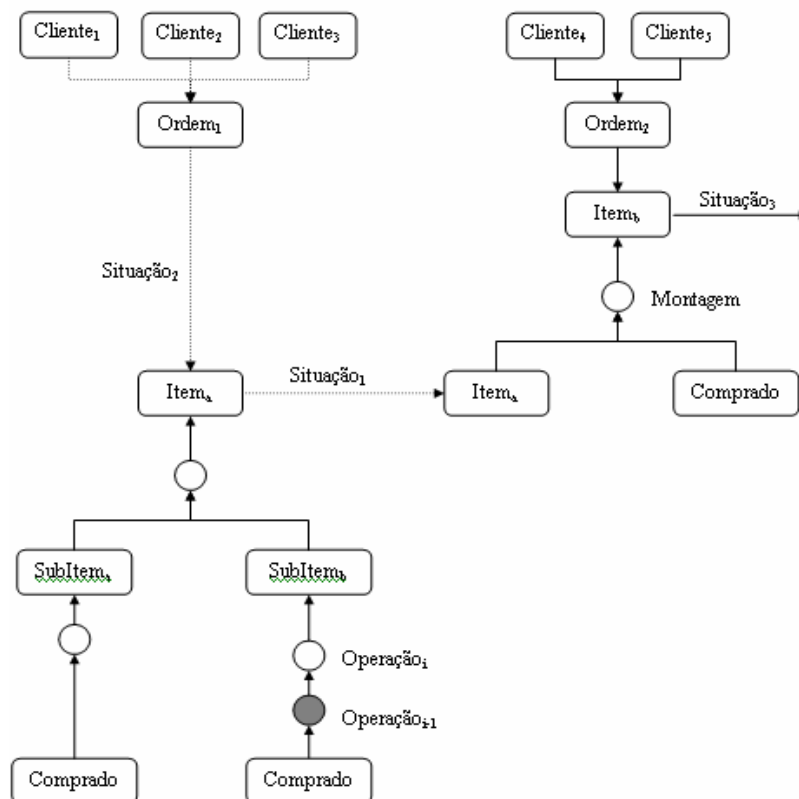


Figura 6.11 Localização de uma operação para cálculo de sua prioridade

Se o item pertence a uma ordem que atende pedidos (situação 2), a sua prioridade é a maior entre estes pedidos, matematicamente tem-se

$w_i = \max\{w_{p1}, w_{p2}, \dots, w_{pn}\}$. Enquanto que se o item atende a uma outra estrutura, sua prioridade será a maior entre todas as estruturas atendidas. Esta análise prossegue recursivamente até se chegar ao nível de ordens (situação 1 e 3).

Outro fator de significância é o status da ordem. Este pode ser firme ou planejado. Firme significa que a ordem já foi liberada para produção e já possui ações para a sua conclusão. O contrário é uma ordem planejada, que será ainda liberada à produção. Logicamente, na maioria dos casos, não é desejável, pelo menos na visão do chão de fábrica, processá-las antes de ordens firmes.

A prioridade conduz o sequenciamento, ditando a importância de uma determinada operação. Quanto maior a prioridade maior a **chance** de ser processada primeiro. Para uma determinada operação, sua prioridade é calculada pela fórmula (6.1), a seguir:

$$w_i = Status \cdot k_1 + fmap(w^*) \quad (6.1)$$

As variáveis de (6.1) são:

w_i (*weight*) : prioridade ou peso da operação i;

$$Status = \begin{cases} 1, & \text{se status da ordem igual a firme;} \\ 0, & \text{qualquer outro caso;} \end{cases}$$

k_1 : uma constante a ser definida, indicando a importância que se deve dar a ordens firmes, aditivamente em relação as ordens planejadas.

w^* é a prioridade do item ou da ordem. Este valor é informado pelo usuário, possuindo um caráter subjetivo e expressa o seu sentimento de importância ou criticidade do item. É calculado conforme explicação dada na figura 6.11. Por questões operacionais, a prioridade é dada da seguinte maneira, 0 é a maior prioridade e 99 é a menor prioridade.

Assim, quanto mais próximo de zero, maior a prioridade do item ou da ordem. Formalmente tem-se:

$$\{w^* \in \mathbb{N} \mid w^* \geq 0 \wedge w^* \leq 99\} \quad (6.2)$$

Para transformar esta prioridade em uma prioridade que tenha o sentido contrário, quanto maior seu valor, maior a prioridade, usa-se a função *fmap*, conforme fórmula (6.3) a seguir:

$$fmap(w^*) = \left(1 - \frac{w^*}{\max(w^*)}\right) \cdot k_2 \quad (6.3)$$

A função de mapeamento $fmap$ converte a prioridade do usuário para o intervalo $[0, k_2]$.

O parâmetro k_2 indica a importância aditiva da prioridade do item ou da ordem.

Para um esclarecimento, a tabela 6.1 fornece um resumo das possibilidades.

Tabela 6.1 Resumo das combinações dos parâmetros

Status de Ordem	Prioridade do cadastro	Prioridade do Sistema
Ordem Firme	0 (maior prioridade para o usuário)	$k_1 + k_2$
Ordem Firme	99 (menor prioridade para o usuário)	k_1
Ordem Planejada	0 (maior prioridade para o usuário)	k_2
Ordem Planejada	99 (menor prioridade para o usuário)	0

Se k_1 for maior que k_2 , sempre se terá a prioridade do sistema de uma ordem firme maior do que qualquer ordem planejada, independente da prioridade do cadastro.

Fica possível perceber que neste ponto existe uma significativa interferência do usuário no algoritmo, tornando o algoritmo suscetível a diferenças de programação, dependente dos critérios subjetivos de cada usuário. Esta permissão é proposital, com o objetivo de flexibilização do algoritmo.

6.4.2 Designação de Ferramental

A figura 6.12 exemplifica o problema a ser resolvido. Existem 4 operações e 2 ferramentas. A pergunta que deve ser respondida é: qual ferramenta é utilizada por qual operação de forma a equilibrar a sua carga de utilização? É apresentada nas seções seguintes uma técnica de Programação Dinâmica Multirotulada, baseada em (Darú,

2005), que efetua o balanceamento do uso dos ferramentais, utilizando como função objetivo um conjunto de 2 normas, do máximo e euclidiana⁶.

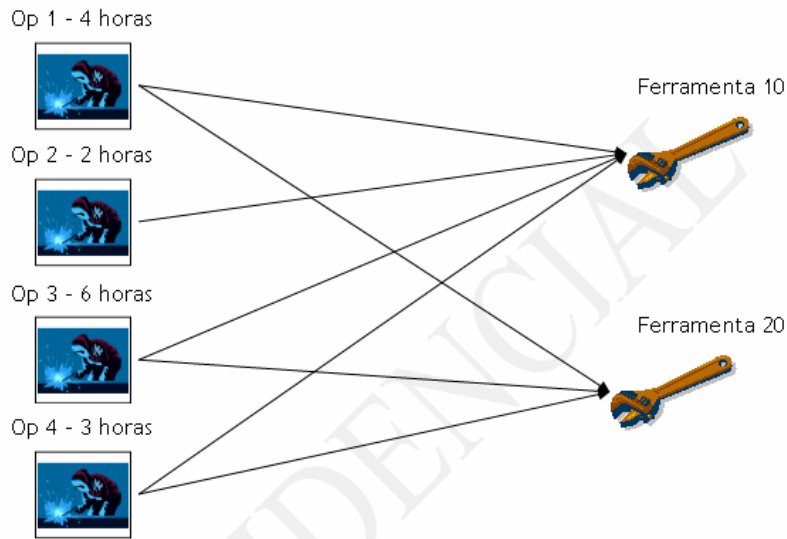


Figura 6.12 Exemplo do problema de balanceamento de ferramental

Considerações sobre a Terminologia

Definição de Programação Dinâmica

A Programação Dinâmica é uma metodologia de solução recursiva de problemas baseada na conjectura de que se o problema puder ser decomposto em partes ou **estágios**, onde estes sejam dependentes apenas dos estágios antecessores (para uma solução *forward*) ou sucessores (solução *backward*), então pode ser resolvido por passos, onde a cada passo, avalia-se a melhor alternativa antecessora com a qual se consegue atingir o ponto em questão.

Variáveis

A figura 6.13 exibe uma representação típica de solução. No eixo x encontram-se as operações e no eixo y as ferramentas (conforme exemplo da figura 6.12). Esta forma permite visualizar de maneira global as soluções que devem ser avaliadas. Os

⁶ **Norma** é definida como $\|x\|: X \rightarrow \mathfrak{R}$ se

- a) $\|x\| \geq 0 \forall x \in X, \|x\| = 0 \Leftrightarrow x = \vec{0}$
- b) $\|x + y\| \leq \|x\| + \|y\|, \forall x, y \in X$
- c) $\|\alpha x\| = |\alpha| \|x\|, \forall \alpha \in \mathfrak{R}, x \in X$

pontos realçados na figura representam uma alternativa de solução. Como exemplo (3,10) (em destaque) indica utilizar o ferramental 10 na operação 3.

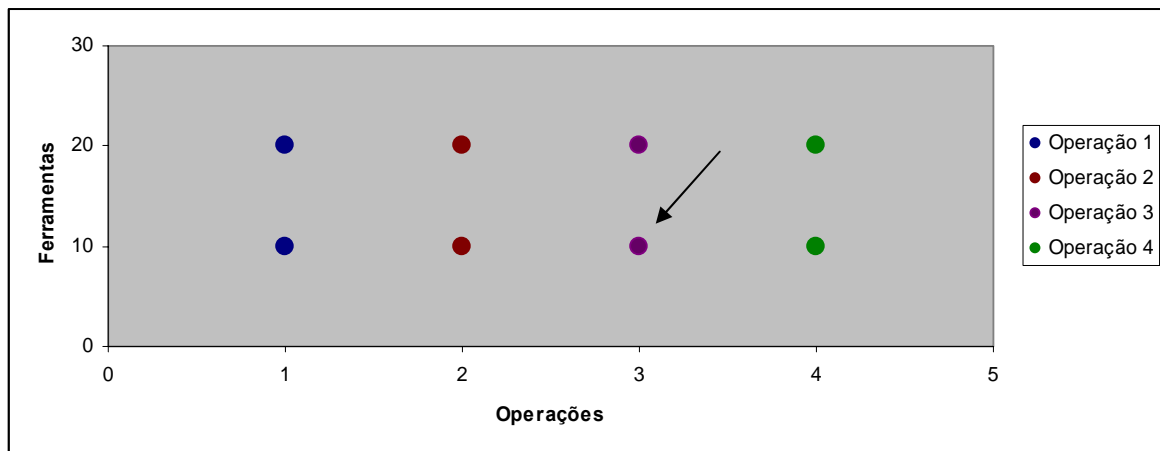


Figura 6.13 Representação das Variáveis

Estágios e Alternativas ou Estados

A representação da figura 6.13 permite definir estágios e alternativas. Os estágios são as etapas que devem ser analisadas individualmente, levando-se em conta apenas o estágio anterior. As alternativas são as opções que se possui para analisar. Assim, os estágios para o exemplo acima são as operações (eixo x) e as alternativas são os ferramentais (eixo y).

Para o caso do algoritmo aqui exposto, trata-se de operações as quais possuem uma data de entrega. A sequência das operações é determinada por sua data de entrega, da menor para a maior, isto objetivando-se equilibrar as cargas conforme as necessidades de entrega vão se tornando iminente.

Rótulo

Outro fator importante na Programação Dinâmica é a função que avalia uma solução no estágio. Para que seja possível identificar qual a melhor alternativa do estágio anterior ou sucessor, cada variável avaliada anteriormente recebe um **rótulo** que determina o valor da função objetivo. É necessário, assim, para cada estágio, analisar um conjunto de soluções e em seguida selecionar a melhor para aquela alternativa. Para exemplificar, as figura 6.14 e figura 6.15 exibem alguns rótulos e avaliam a solução de um estágio.

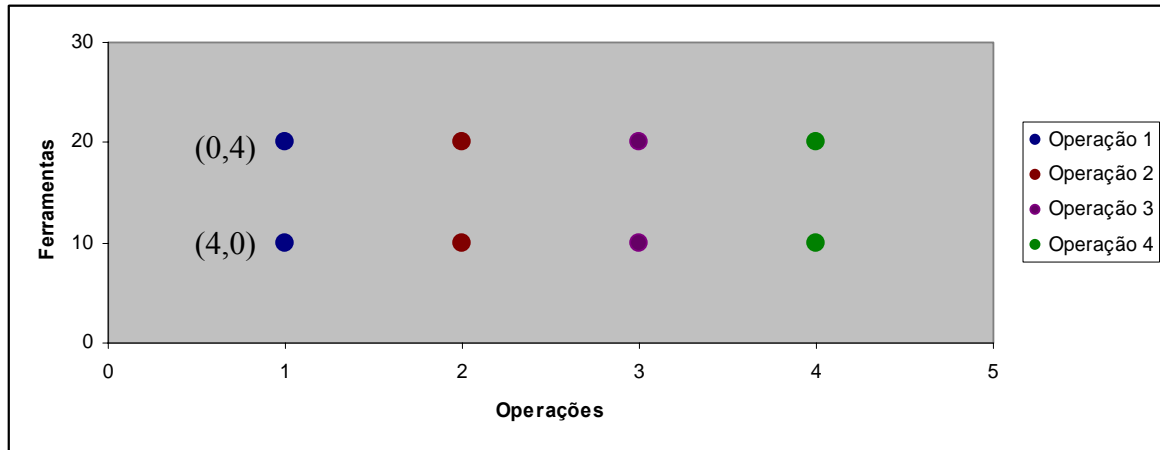


Figura 6.14 Exemplo de rótulo

Um rótulo, para este exemplo corresponde a um vetor de utilização total de cada ferramenta. Este vetor pode ser escrito como $t = (t_1, t_2)$, onde t_1 corresponde ao tempo total de utilização do ferramental 10 e t_2 do ferramental 20. Da figura 6.14, pode-se observar que o rótulo associado ao ponto (1,10) é (4,0) que diz o seguinte: ao se utilizar na operação 1 a ferramenta número 10 tem-se uma ocupação associada de 4 horas para a ferramenta 10 e 0 para a ferramenta 20. Da mesma forma o rótulo associado ao ponto (1,20) é (0,4).

Para avaliar uma alternativa, verifica-se todas as possibilidades do estágio anterior. Conforme pode ser visto pela figura 6.15, para chegar-se a posição (2,10) pode-se partir de (1,10) ou de (1,20), isto é, é possível utilizar a ferramenta 10 **ou** 20 na operação 1. Assim, tem-se duas possibilidades, alocar a ferramenta 10 ou ferramenta 20. Se for proveniente da ferramenta 10 tem-se uma ocupação de 6 horas (pois foram utilizadas 4 horas para a primeira mais 2 horas para a segunda operação), enquanto que se provier da ferramenta 20 tem-se 2 horas de utilização da ferramenta 20 e 4 horas da ferramenta 10.

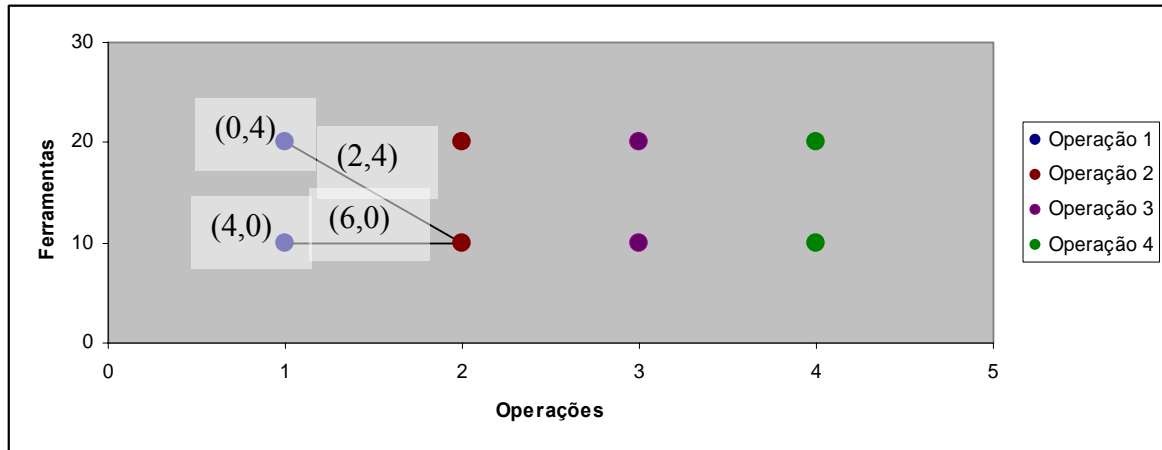


Figura 6.15 Avaliação de alternativa

Como se pode perceber o rótulo consiste de um vetor de ocupação, mas em grande parte dos problemas de Programação Dinâmica, um número apenas avalia a decisão. Quando se tem um vetor como rótulo, dá-se o nome de multirótulo. A questão que surge é de que forma avaliar qual o melhor vetor? Qual o melhor rótulo? O próximo tópico trata do critério de avaliação.

Função de Avaliação

Para avaliação da melhor alternativa, utiliza-se uma função que mensura o quão “boa” é uma solução. Para o caso de multirótulo, tem-se um vetor e assim, utiliza-se uma norma para avaliação. Sendo o vetor $t = (t_1, t_2, \dots, t_n)$, as normas utilizadas são:

1. Norma do máximo ou infinito, definida como:

$$\|t\|_{\infty} = \max(t_1, t_2, \dots, t_n) \quad 6.4$$

2. Norma euclidiana, definida como:

$$\|t\|_2 = \left(\sum_{i=1}^n t_i^2 \right)^{\frac{1}{2}} \quad 6.5$$

Devido a facilidade de avaliação, ou diminuição do tempo de cálculo, a raiz não é necessária, logo a norma pode ser avaliada por:

$$\|t\|_2 = \sum_{i=1}^n t_i^2 \quad \text{Equação 6.6}$$

Para definir o critério de seleção, desenvolve-se primeiro o conjunto de vetores $T = (t_1, t_2, \dots, t_n)$, onde t_i é um vetor. Assim o critério de seleção é:

Algoritmo 6.1 Avaliação de AlternativaAlgoritmo Avaliação de AlternativaVariáveis de Entrada T : Conjunto de vetores, referentes as alternativas a serem avaliadas n : Número de alternativasVariáveis de Saída A : Alternativa selecionadaInícioAtribuir a A a alternativa 1Atribuir a Norma do máximo da Alternativa (NMA) o valor $\|t_1\|_\infty$ Atribuir a Norma euclidiana da Alternativa (NEA) o valor $\|t_1\|_2$ Para[1] i de 2 até n FaçaSe[1] $\|t_i\|_\infty < NMA$ EntãoAtribuir a A a alternativa 1Atribuir a Norma do máximo da Alternativa (NMA) o valor $\|t_i\|_\infty$ Atribuir a Norma euclidiana da Alternativa (NEA) o valor $\|t_i\|_2$ SenãoSe[2] $\|t_i\|_\infty = NMA$ EntãoSe[3] $\|t_i\|_2 < NEA$ EntãoAtribuir a A a alternativa 1Atribuir a Norma do máximo da Alternativa (NMA) o valor $\|t_i\|_\infty$ Atribuir a Norma euclidiana da Alternativa (NEA) o valor $\|t_i\|_2$ Fim Se[3]Fim Se[2]Fim Se[1]Fim Para[1]Fim

Como exemplo, sejam os vetores $t_1=(1,3)$ e o vetor $t_2=(2,2)$, calcula-se a norma do máximo, e encontra-se:

$$\|t_1\|_\infty = \|(1,3)\|_\infty = 3 \text{ e } \|t_2\|_\infty = \|(2,2)\|_\infty = 2$$

Seleciona-se então o vetor t_2 .

Como outro exemplo, sejam $t_1=(1,3)$ e o vetor $t_2=(2,3)$, calcula-se a norma do máximo, e encontra-se:

$$\|t_1\|_\infty = \|(1,3)\|_\infty = 3 \text{ e } \|t_2\|_\infty = \|(2,3)\|_\infty = 3$$

Como ocorreu empate, calcula-se a norma euclidiana.

$$\|t_1\|_2 = \|(1,3)\|_2 = \sqrt{10} \text{ e } \|t_2\|_2 = \|(2,3)\|_2 = \sqrt{13}$$

Assim, o vetor escolhido é t_1 .

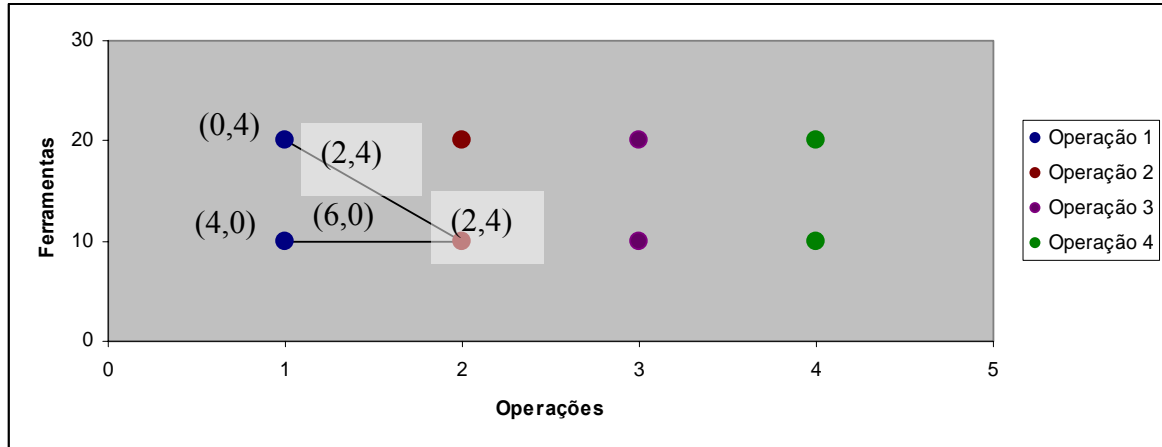


Figura 6.16 Exemplo de avaliação de alternativas

Como exemplo de aplicação, para a figura 6.16 tem-se os vetores $t_1=(2,4)$ e o vetor $t_2=(6,0)$. Calculando a norma do máximo encontra-se:

$$\|t_1\|_\infty = \|(2,4)\|_\infty = 4 \text{ e } \|t_2\|_\infty = \|(6,0)\|_\infty = 6$$

Escolhe-se assim, o menor valor entre 4 e 6. A solução escolhida é a opção 1, rótulo (2,4). Isto é, no caso de se escolher a ferramenta 10 para a operação 2, deve-se escolher a ferramenta 20 para a operação 1, conforme rótulo acima do ponto (2,10).

Trajectoria

A questão agora é como recuperar o caminho. Onde se encontra a informação que denota de onde é proveniente a solução ou qual foi o vetor escolhido dentre todas as alternativas. Novamente com relação ao problema das operações e ferramental, qual foi a ferramenta escolhida para se utilizar com determinada operação. Para que se possa responder a esta pergunta, durante o processo de análise gera-se uma matriz conhecida como matriz de trajetória, a qual permite recuperar todas as decisões tomadas nos passos intermediários.

Por definição

$W = \{w_{ij}: i = 1, 2, \dots, \text{número de ferramentas e } j = 2, \dots, \text{número de operações}\}$
 onde $w_{i1} = \emptyset$, pois não há caminho para a 1ª operação, e w_{ij} representa o seguinte: ao se colocar a ferramenta i para a operação j , colocou-se na operação $j-1$ a ferramenta w_{ij} . Um exemplo de W será mostrado a frente.

Exemplo 6.1 Aplicação do algoritmo para a figura 6.12

Seja um conjunto de operações $O = \{o_1, o_2, o_3, o_4\}$ e um conjunto de ferramentas $F = \{f_1, f_2\}$, que correspondem ao problema da figura 6.12. A matriz P é dada por:

$$P = \begin{matrix} & \begin{matrix} f_1 & f_2 \end{matrix} \\ \begin{matrix} o_1 \\ o_2 \\ o_3 \\ o_4 \end{matrix} & \begin{bmatrix} 4 & 4 \\ 2 & \infty \\ 6 & 6 \\ 3 & 3 \end{bmatrix} \end{matrix}$$

O valor ∞ informa que não é possível utilizar a ferramenta 10 (f_1) na operação 2, conforme pode ser visto em figura 6.12. Ou, ao utilizar esta, ficará infinitamente sendo consumida por esta operação. Como o objetivo é minimizar a carga, esta solução nunca será adotada.

Como um exemplo de aplicação, é mostrado na figura 6.17, o caminho de solução para o problema da figura 6.12. O primeiro estágio é o valor do tempo de utilização da ferramenta i , pela operação j na posição i , em resumo tem-se $\{(1,10):(4,0)\}$ e $\{(1,20):(0,4)\}$, donde o primeiro vetor é o ponto no gráfico e o segundo vetor é o tempo de uso das ferramentas. A notação aqui adotada foi $\{(\text{operação, ferramenta}):(\text{carga ferramenta1, carga ferramenta 2})\}$. Ao passar para o segundo estágio, tem-se que avaliar alternativas. Verifica-se duas possibilidades para se chegar ao ponto (2,10), provir de $\{(1,10),(6,0)\}$ ou de $\{(1,20),(2,4)\}$. Tomando a norma do máximo das duas opções, vê-se que o candidato $\{(1,20),(2,4)\}$ é o menor. Já para o estágio três não há alternativa para ser avaliada e os rótulos ficam, $\{(3,10):(8,4)\}$ e $\{(3,20):(2,10)\}$. Por fim, para o quarto estágio, avalia-se duas alternativas para o ponto (4,10), a saber $\{(3,10):(11,4)\}$ e $\{(3,20):(8,10)\}$ sendo escolhido o ponto (3,20). Para o ponto (4,20) tem-se $\{(3,10):(8,7)\}$ e $\{(3,20):(2,13)\}$, donde se escolhe (3,10). A figura 6.17 exhibe as trajetórias avaliadas.

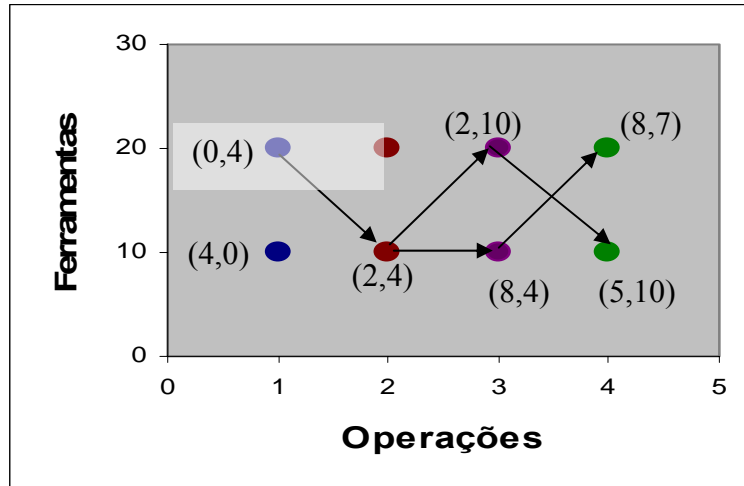


Figura 6.17 Aplicação do Algoritmo de designação de ferramental para a figura 6.12

Recuperação da Trajetória

Tomando como base a figura 6.17, o vetor trajetória é dado por:

$$W = \begin{bmatrix} - & - & 10 & 10 \\ - & 20 & 10 & 20 \end{bmatrix}$$

Para encontrar a melhor solução aplica-se aos rótulos do último estágio o critério de decisão. Avaliando-se r_{41} e r_{42} , tem-se o seguinte:

$$\|(8,7)\|_{\infty} = 8$$

$$\|(5,10)\|_{\infty} = 10$$

Como a norma de (8,7) é menor que norma de (5,10), escolhe-se o rótulo (8,7) ou seja, ponto (4,20). Desta forma, recupera-se a trajetória da seguinte maneira.

Ferramenta da operação 4 = 20

$w_{42} = 10$, logo

Ferramenta da operação 3 = 10, pois para chegar a posição (4,20) escolheu-se a ferramenta 10.

$w_{31} = 10$, assim, a ferramenta escolhida é a 10 e por fim $w_{21} = 20$. A ferramenta é a 20. Um resumo da utilização fica:

$$\begin{array}{l} \text{operação} \begin{bmatrix} 1 & 2 & 3 & 4 \end{bmatrix} \\ \text{ferramenta} \begin{bmatrix} 20 & 10 & 10 & 20 \end{bmatrix} \end{array}$$

A trajetória final para o problema da figura 6.12 é exposta pela figura 6.18, a seguir.

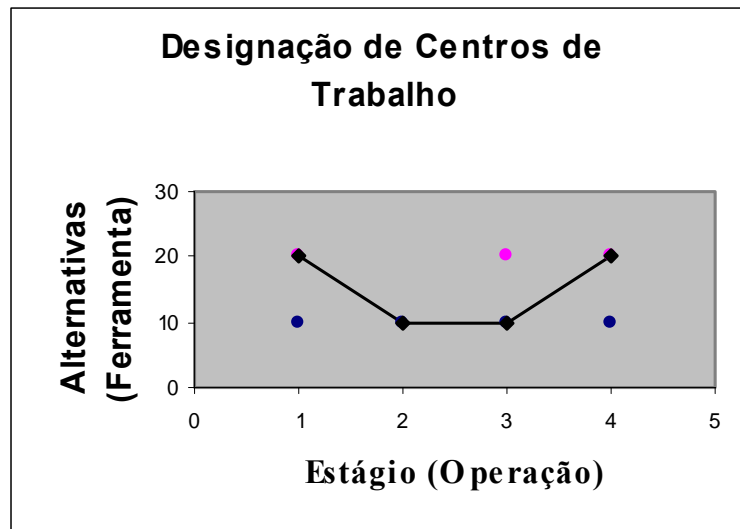


Figura 6.18 Trajetória final para o problema da figura 6.12

A designação final é exibida pela figura 6.19, a seguir:

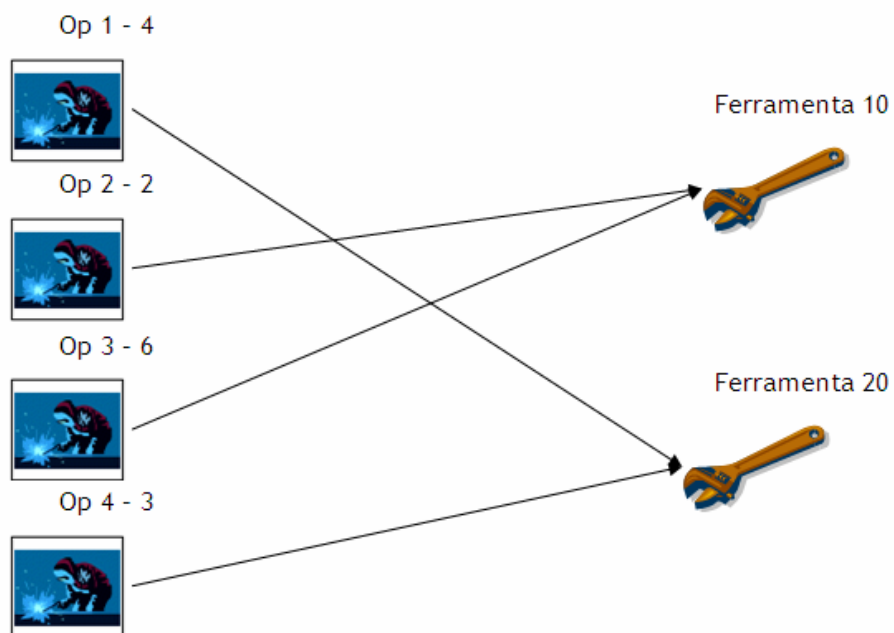


Figura 6.19 Designação após execução do algoritmo

O algoritmo acima não respeita o princípio de Bellmann (Bellmann, 1974) e portanto não garante a solução ótima. Comentários adicionais podem ser vistos em (Darú, 2005) sobre a aplicação deste algoritmo.

6.4.3 Designação de Centros de Trabalho

Para a alocação dos centros de trabalho, foram levantadas na prática duas situações desejáveis. A primeira informa que se deve alocar para uma determinada

operação o centro de trabalho preferencial. Isto é, o recurso que tem a maior velocidade de processamento ou o que foi adaptado por motivos operacionais, ou ainda o que a empresa deseja utilizar por motivos estratégicos. Enquanto a segunda tem como objetivo o bom balanceamento e uso dos recursos ou centros de trabalho. A primeira foi resolvida, mas não será objeto de apreciação neste trabalho, enquanto que a segunda é descrita na seção seguinte.

Alocação Baseada em Balanceamento Uniforme

O objetivo desta formulação é balancear a carga de utilização dos centros de trabalho utilizados. Entenda-se por balanceamento uma distribuição o mais igual possível (em percentual), dentro do horizonte de tempo estimado, para cada centro de trabalho considerado. Para resolução do problema de alocação do centro de trabalho com o objetivo de balanceamento uniforme foi utilizada uma técnica similar ao da designação de ferramental, com quatro diferenças:

1. Leva-se em consideração a velocidade de processamento do centro de trabalho, diferentemente do ferramental;
2. Leva-se em consideração a disponibilidade do centro de trabalho;
3. Leva-se em conta a utilização das ferramentas;
4. Levam-se em conta as datas mínimas ou mais cedo de cada operação;
5. Operações que possuem a mesma ferramenta ficam no mesmo centro de trabalho.

Para uma descrição completa do algoritmo de designação, faz-se necessário a definição de algumas variáveis bem como sua representação. Estas definições são vistas nos tópicos seguintes.

Estágios e Alternativas

Para a designação dos CT, o estágio foi definido como sendo as tarefas, e as alternativas como sendo os centros de trabalho. Assim o par ordenado (x,y) representa, designar para a tarefa (x) o centro de trabalho (y) . A figura 6.20 exhibe esta representação.

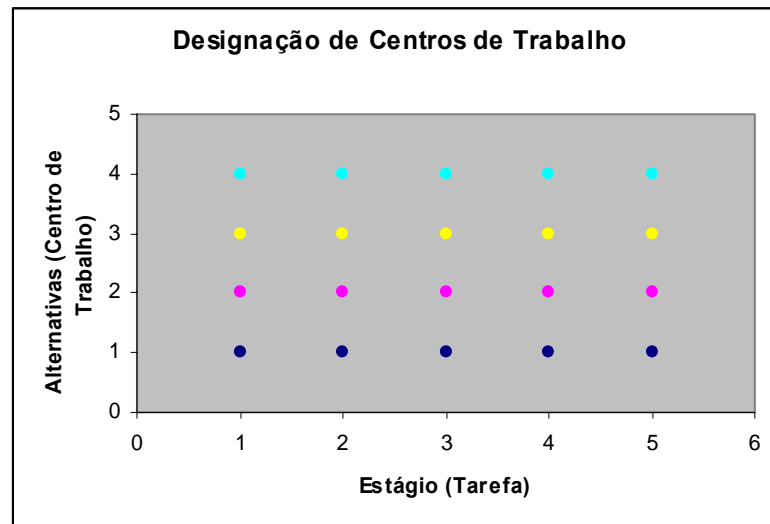


Figura 6.20 Exemplo de definição de estágio e alternativa para designação de ferramental

Assim como nas ferramentas, nem todos os centros de trabalho são permitidos para processar as operações. Diferentemente das ferramentas, onde o tempo de utilização era constante, para os centros de trabalho estes são diferentes. Estas diferenciações são tratadas a seguir.

Centros de Trabalho Válidos para Operação

Um CT(Centro de Trabalho) é um recurso com capacidade de processar uma determinada tarefa. Porém, devido as características tecnológicas, algumas tarefas devem ou só podem ser executadas em alguns centros de trabalho. Isto pode ser visualizado, por exemplo, informando que a operação dois só pode ser processada pelo CT dois, e a operação quatro, pelos centros dois e quatro. Esta nova situação é exibida pela figura 6.21.

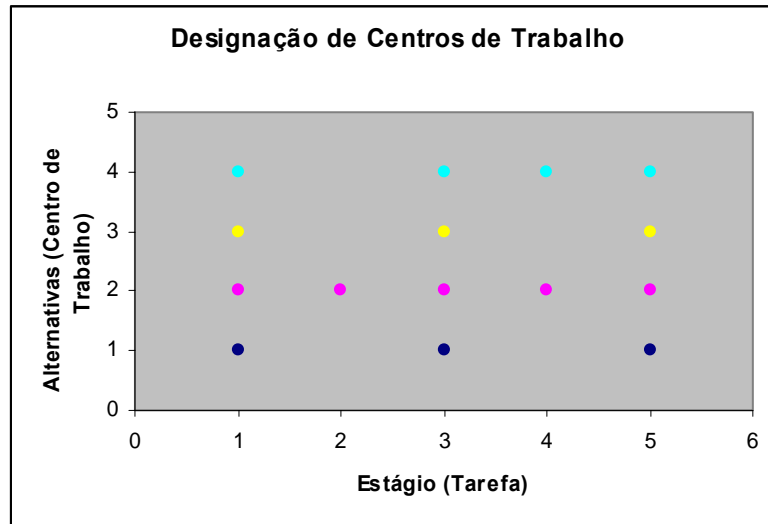


Figura 6.21 Exemplo com a inclusão da restrição de centros de trabalho válido

Performance dos Centros de Trabalho

Da mesma forma que um centro pode ou não ser permitido processar uma operação, pode-se ter situações em que o tempo para se processar uma operação seja diferente dependendo do centro utilizado.

Seja o tempo padrão p_i definido como o tempo necessário para se processar uma determinada operação, tomando-se como base o centro convencionado como padrão. Desta forma a performance é dada como um percentual em termos da velocidade de processamento. O tempo que um centro leva para processar uma operação é dada pela fórmula (6.7)

$$p_{ij} = \frac{p_i}{v_{ij}} \quad (6.7)$$

Onde

p_{ij} = tempo do centro de trabalho j para processar a operação i ;

p_i = tempo de processamento padrão;

v_{ij} = velocidade (em %) do centro de trabalho j para processar a operação i .

Para um exemplo, sejam 3 centros de trabalho diferenciados por características tecnológicas, e suas *performances* supostamente constantes para simplificação, conforme figura 6.22, a seguir:




Fresadora CNC	Fresadora CN	Fresadora Manual
		
Tempo Padrão (TP)	50% (2 x TP)	33% (3 x TP)

Figura 6.22 Centros de trabalho e suas performances

Sejam ainda cinco operações conforme tabela 6.2, a seguir:

Tabela 6.2 Tempos padrão das operações

Operação	1	2	3	4	5
Tempo Padrão	5	9	7	2	4

Como um exemplo, abaixo exibe-se o cálculo do tempo de processamento da operação quatro (duas unidades de tempo) nos centros de trabalho dois (CN com 50%) e três (manual, com 33 por cento).

Fresadora CN	Fresadora Manual
$Tempo = \frac{2}{0,5} = 4$	$Tempo = \frac{2}{0,33} = 6$

A seguir o tempo de processamento para cada operação em cada centro de trabalho é apresentado na tabela 6.3:

Tabela 6.3 Tempo Processamento Operação x Centro de Trabalho

Centro	CNC	CN	Manual
Operação	Tempo	50%	33%
1	5	10	15
2	9	18	27
3	7	14	21
4	2	4	6
5	4	8	12

Depois de executado o algoritmo de balanceamento conforme descrito para o balanceamento de ferramental, encontra-se a seguinte resposta, ilustrada pela figura 6.23:

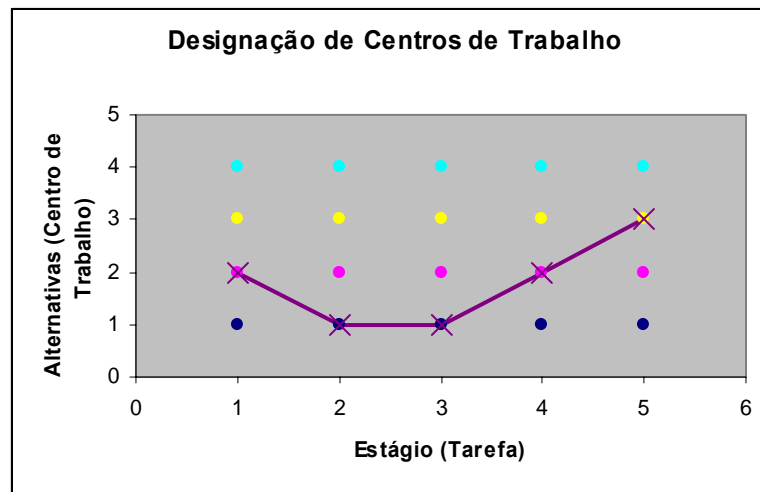


Figura 6.23 Solução do exemplo, considerando performance dos centros de trabalho

A tabela abaixo resume a solução, explicitando os tempos de processamento e o total de utilização por centro de trabalho.

Tabela 6.4 Modelo tabular para solução do exemplo

C3					12	12
C2	10			4		14
C1		9	7			16
	O1	O2	O3	O4	O5	Total

Disponibilidade dos Centros de Trabalho

Um problema comum no chão de fábrica é a disponibilidade das máquinas. Nem todos os centros estão disponíveis todo o tempo. E devido a isto, a métrica adotada para se determinar a melhor solução já não é mais adequada. Isto porque se tivermos um centro com baixa disponibilidade, devemos ocupar a carga dos demais, mesmo que o tempo de processamento acabe sendo maior.

Definindo a carga do centro como sendo a razão entre o tempo total utilizado do centro e sua disponibilidade, o objetivo é então selecionar a solução que possui a menor carga máxima utilizada. O rótulo sofre uma leve alteração, não sendo mais definido

como a ocupação, mas sim esta ocupação dividida pela disponibilidade do centro. O novo valor t^* é definido pela fórmula (6.8).

$$t_i^* = \frac{t_i}{d_i} \quad (6.8)$$

Onde

t_i = tempo total utilizado do centro de trabalho i ;

d_i = disponibilidade do centro de trabalho i .

E o rótulo é aplicado sobre o mínimo da norma dos vetores t^* , conforme as fórmulas (6.4) e (equação 6.6).

Supondo, como exemplo, que por algum motivo a fresadora CNC (C1) terá uma parada de 2 dias, a máquina fresadora CN (C2) terá uma parada de 1 dia. Assim, a programação agora avalia o percentual de ocupação de cada centro de trabalho em relação a sua disponibilidade e o vetor disponibilidade é dado por (8,16,24) em horas, conforme tabela abaixo.

Tabela 6.5 Disponibilidade dos os centros de trabalho

Centro	CNC	CN	Manual
Disponibilidade	8	16	24

Executando o algoritmo com a diferença que agora é analisado o vetor percentual de utilização e não mais o número de horas. Tem-se a resposta ilustrada pela figura 6.24:

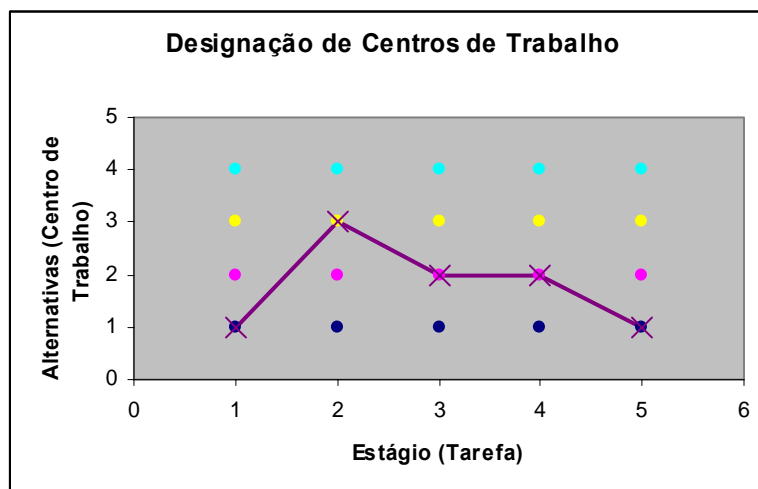


Figura 6.24 Solução considerando disponibilidade dos centros

A tabela 6.6 resume os resultados em forma tabular. Pode-se perceber que o objetivo foi atendido, ou seja, os centros de trabalho possuem percentuais de utilização idênticos. A nova função objetivo conduziu a uma solução totalmente diferente, devido a mudança da forma de cálculo do rótulo. Lembrando que decisões relativas a *performance* dos centros e suas disponibilidades foram levadas em consideração.

Tabela 6.6 Modelo tabular para solução do exemplo considerando disponibilidades

C3	27					27	24	112,5
C2	14 4					18	16	112,5
C1	5	4				9	8	112,5
	O1	O2	O3	O4	O5	Total	Disponibilidade	%Utilização

Ferramental Acoplado

Na maioria dos casos, uma troca de ferramentas não é uma boa prática de programação da produção e deve-se evitar. Sendo assim, o algoritmo aloca operações que usam a mesma ferramenta no mesmo recurso. Como regra geral: se usa a mesma ferramenta então usa o mesmo recurso. O propósito é minimizar o *setup* entre trocas de ferramentas.

Tomando-se como base os tempos de processamento da tabela 6.2 e aplicando o algoritmo de balanceamento de ferramental com apenas duas ferramentas, obtém-se a solução exposta na tabela 6.7.

Tabela 6.7 Modelo tabular para solução do exemplo balanceando ferramentas

F2	9 4 13				
F1	5	7 2 14			
	O1	O2	O3	O4	O5

Estes resultados são enviados ao alocador de recursos. O alocador de recursos, por sua vez, cria blocos de operações. Estes blocos são justamente as operações de cada ferramenta. Assim, os blocos criados conforme exemplo são: operações (1,3,5) e Operações (2,4). Uma vez estes criados, o alocador ao atribuir um centro de trabalho o faz levando-se em conta o bloco inteiro. É como se tivéssemos uma operação de 13 e outra de 14 unidades de tempo.

A justificativa de se fazer isto é devido ao fato que primeiro, o tempo de *setup* impossibilita a troca constante de recurso pela ferramenta, e segundo que a operação deve esperar a conclusão de sua antecessora que usa o mesmo ferramental.

Como exemplo, utilizando agora os mesmos três centros, com as performances conforme exibido na figura 6.22 e disponibilidades dadas pela tabela 6.5 tem-se a seguinte solução, ilustrada através da figura 6.25:

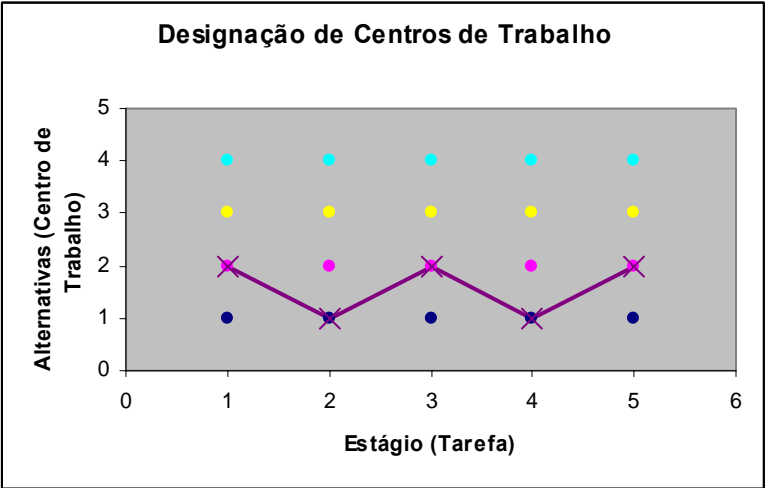


Figura 6.25 Solução considerando-se ferramental

Conforme pode ser visto, o algoritmo procurou respeitar o balanceamento, levando-se em conta a *performance* e a disponibilidade, além de impor como restrição de ferramentas. A tabela 6.8 detalha o resultado obtido.

Tabela 6.8 Modelo tabular para solução do exemplo considerando ferramenta

C3						0	24	0
C2	10		14		8	28	16	175
C1		9		4		13	8	162
						O1	O2	O3
						O4	O5	Total
						Disponibilidade		%Utilização

Comentários Sobre a Solução

Esta solução acaba por não usar o centro de trabalho três, pois como foi exposto, tem-se a questão do ferramental, da performance e da disponibilidade.

Data Mínima

Outro problema freqüente é a data mínima, a qual determina qual o instante em que a operação está disponível para ser executada. Isto porque devido a operações predecessoras e disponibilidade de material, esta operação só poderá ser executada quando a data mínima for atingida. Além disto, devido ao cálculo de conversão de calendário a data mínima é dependente do centro de trabalho. Desta forma, é necessário redefinir o cálculo de t . Deve-se ter o cuidado de respeitar esta data conforme o centro de trabalho, sendo considerado para efeitos de designação, o maior valor entre a somatória dos tempos utilizados ou a data mínima. O algoritmo também é modificado para ordenar as operações por data mínima e os estágios (tarefas) respeitam esta relação de ordem.

O algoritmo, da mesma forma que o executado para designação do ferramental, não tem garantia de otimalidade, devido a não respeitar o princípio de Bellmann (Bellmann, 1974).

6.4.4 Sequenciamento de Tarefas

Neste estágio do sistema, o objetivo passa a ser identificar a ordem em que as tarefas devem ser processadas, realizando tal tarefa para cada centro de trabalho.

A dificuldade do problema de sequenciamento já foi discutida na seção 5.6, devido a isto, o problema foi desmembrado nas fases listadas a seguir:

- a) Utilização da *TOC*, com o objetivo de podar o domínio e concentrar-se apenas nos centros de trabalho restritivo. O fato de outros centros tornarem-se restritivos após o sequenciamento é contornado pela utilização do pulmão de restrição, impedindo assim que o recurso crítico sofra paradas ou não cumpra a programação.
- b) Designação de ferramental, para realizar um “agrupamento” das operações por ferramenta.
- c) Designação de centros de trabalho, visando deixar juntas as operações que usam o mesmo ferramental e equilibrar a sua utilização visualizando sua *performance* e disponibilidade ou simplesmente alocando conforme prioridade do centro de trabalho efetuar uma dada operação.

Após a execução destas fases, tem-se definido para cada operação, qual ferramenta esta pode utilizar e em qual centro esta operação deve ser processada. O passo seguinte é determinar a ordem.

A Heurística ATC Modificada

O primeiro passo para efetuar o sequenciamento é definir o objetivo. Um objetivo comum no ambiente operacional relaciona prioridade com atraso. Portanto, o objetivo principal é minimizar o atraso dos pedidos prioritários, isto é, $\min wT$.

A heurística mais indicada na literatura para minimização do wT chama-se Custo Aparente do Atraso (*ATC - Apparent Tardiness Cost*). Foi desenvolvida por (Rachamadugu, 1982), e reescrita nas equações (6.9), (6.10) e (6.11) em suas formas originais:

$$ATC = \frac{w_i}{p_i} \left\{ 1 - \frac{(d_i - p_i - t)^+}{k\bar{p}} \right\}^+ \quad (6.9)$$

$$ATC = \frac{w_i}{p_i} \left\{ 1 - \frac{\bar{p}}{\bar{p} + k(d_i - p_i - t)^+} \right\} \quad (6.10)$$

$$ATC = \frac{w_i}{p_i} \exp \left\{ -\frac{k}{\bar{p}} (d_i - p_i - t)^+ \right\} \quad (6.11)$$

Onde lê-se X^+ , substituir por máximo (0,X)

As variáveis e parâmetros utilizados nas equações anteriores são:

w_i = Prioridade da operação i , conforme calculado na seção 6.4.1;

p_i = Tempo de processamento da operação i ;

d_i = Data de entrega da operação i ;

t = Instante de análise, em unidades temporais válidas do centro de trabalho;

k = Parâmetro informado, dependente do cenário produtivo;

\bar{p} = Tempo de processamento médio das operações restantes, a serem processados no centro de trabalho.

Esta heurística tem como principais características ser um híbrido entre as heurísticas que visam entregar as tarefas respeitando suas datas de entrega (data de entrega mais cedo primeiro, *EDD - Early Due Date*), devido a expressão $(d_i - p_i - t)$,

que pode ser rebatizada de folga, e a partir do instante que este valor zera, ou seja, a operação atrasa, o critério passa a ser entregar as operações com maior prioridade por unidade de tempo $\frac{w_i}{p_i}$ ou maior prioridade por unidade de tempo de processamento primeiro (*WSPT – Weighed Shortest Processing Time*).

Além disto, possui a característica de ser uma regra de despacho dinâmica, isto é, a cada iteração é calculado o valor *ATC* para cada tarefa e realizada uma ordenação por este critério. A tarefa que neste instante recebe o maior valor *ATC* é a tarefa escolhida para ser processada.

Para avaliar o comportamento do *ATC* em função de seus parâmetros, esboça-se a figura 6.26. No eixo *x*, tem-se o valor $(\%di-pi)$, que corresponde ao passar do tempo *t* em função do percentual do valor *di-pi*, ou seja $t = x \cdot (di-pi)$. Quando $x = 0$, $t = 0$; quando $x = 1$, $t = di-pi$ e o valor $di-pi-t = 0$. Já no eixo *y*, tem-se o valor calculado do *ATC*, que deve ser multiplicado como um percentual sobre o valor de wi/pi . O parâmetro *k*, influencia de maneira tal, que se este tende a zero ou seja $k \rightarrow 0$, o *ATC* é dependente apenas de wi/pi , enquanto que se $k \rightarrow \infty$, o *ATC* é praticamente constante e igual a zero até $t = di-pi$, após este valor, a dependência é novamente apenas relativa ao valor wi/pi . Na literatura, conforme indicado em (Rachamadugu, 1982), recomenda-se um valor de $k = 3$ para planejamentos dinâmicos ou seja, que possuam muitas tarefas para serem processadas em pouco espaço de tempo, enquanto que (Valente, 2003), utiliza relações dependentes do número de operações e de um fator *TF (Tardiness Factor)*, definido como $1-d/C_{max}$ onde *d* é a média da datas de entrega e C_{max} é a maior data real de entrega, com a diferença que o parâmetro *k'* de (Valente, 2003) é relacionado como $1/k$, em relação ao adotado neste trabalho. Para questões práticas, este trabalho utiliza $k \in [0.5, 3.0]$, ficando como sugestão para trabalhos futuros a adequada escolha deste parâmetro, fazendo sentido a dependência deste com o cenário, conforme dito por (Valente, 2003).

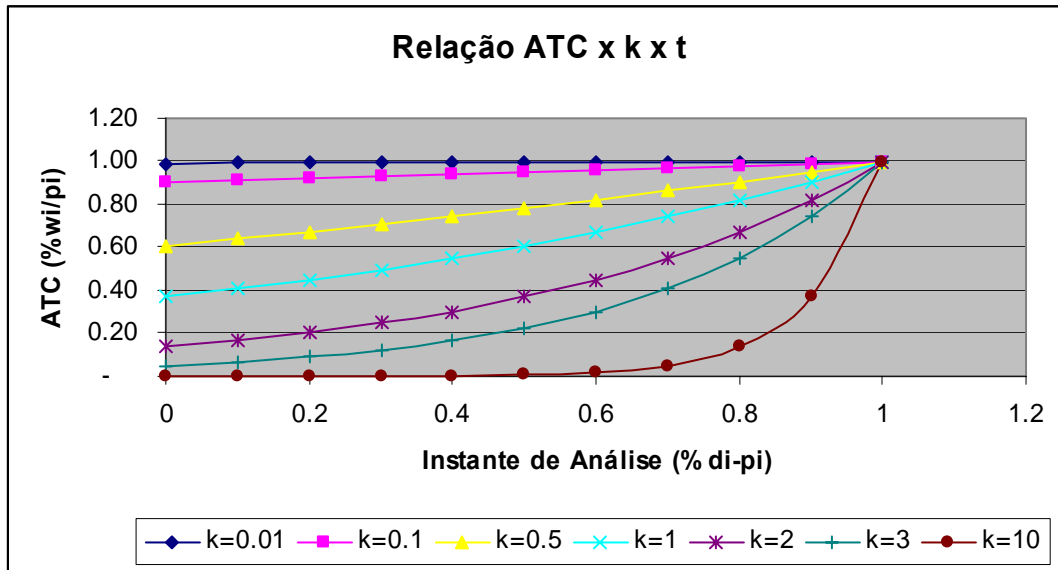


Figura 6.26 Relação de ATC com seus parâmetros

Por questões práticas, o critério w_i/p_i , não é aderente ao meio industrial, isto porque, suponha-se duas operações uma com $w_1 = 100$, $p_1 = 10$ e a outra com $w_2 = 50$ e $p_2 = 4$. Em uma empresa, se uma ordem é prioritária, não é possível explicar para o um usuário que por questões de otimalidade de objetivo que ele deve fazer a operação dois porque ela tem uma prioridade em relação ao tempo maior, ainda mais se a operação o_1 está atrasada e o_2 não. Por isto, a formulação foi modificada para, em caso de atraso, continuar aumentando a prioridade. Esta alteração é refletida em (6.12), extraíndo-se a condição $(d_i - p_i - t)^+$ para $(d_i - p_i - t)$.

$$ATC = \frac{w_i}{p_i} \exp \left\{ -\frac{k}{\bar{p}} (d_i - p_i - t) \right\} \quad (6.12)$$

Isto acarreta não mais levar em consideração, em caso de atraso, somente w_i/p_i , mas sim um acréscimo da prioridade conforme a operação atrasa.

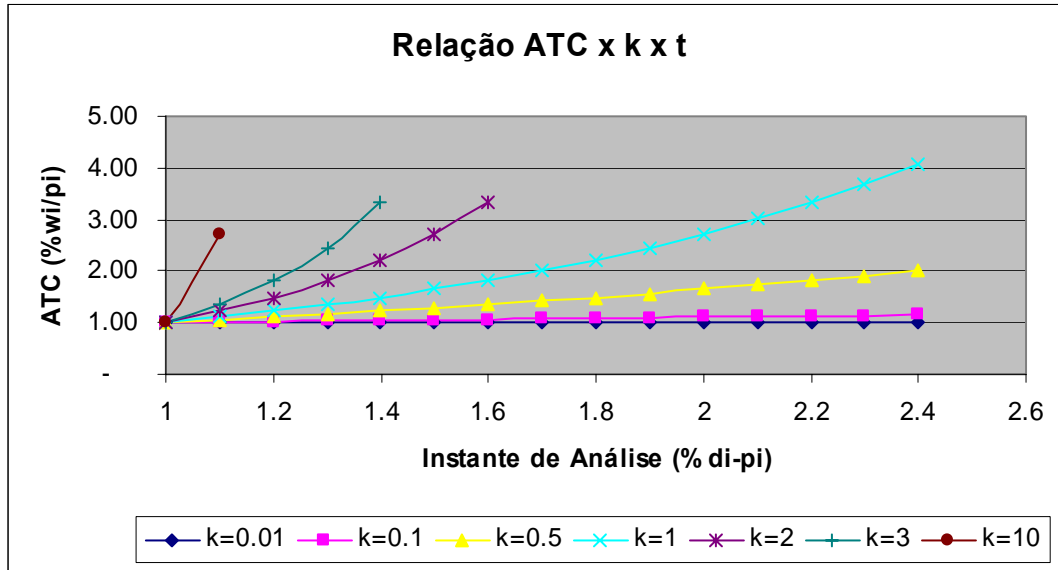


Figura 6.27 Relação ATC com seus parâmetros, porém considerando o atraso.

A figura 6.27 exibe o comportamento do incremento do ATC , conforme o passar do tempo e em função do parâmetro k . É possível observar que quanto maior o parâmetro k , menos tolerância a regra terá de permitir o atraso de uma operação, porém podendo desprezar de tal maneira a prioridade w_i/p_i .

Outra alteração é apenas por questões de nomenclatura. O valor $(d_i - p_i - t)$, nada mais é que a folga de uma operação, que pode ser positiva em caso de antecipação ou negativa em caso de atraso. Assim (6.12) pode ser reescrita como (6.13)

$$ATC = \frac{w_i}{p_i} \exp \left\{ -k \frac{\text{Folga}}{\bar{p}} \right\} \quad (6.13)$$

Por questões de análise, é possível inferir que o valor folga/\bar{p} representa o número médio de operações que ainda podem passar a sua frente (folga positiva) ou o número de operações que já passaram a sua frente (folga negativa). Substituindo folga/\bar{p} por \overline{Nop} , obtém-se (6.14):

$$ATC = \frac{w_i}{p_i} \exp \left\{ -k \cdot \overline{Nop} \right\} \quad (6.14)$$

Esta análise é feita, porque em um ambiente operacional o tempo não flui harmoniosamente; mas sim em pacotes discretos, isto é, em operações. A cada vez que se decide realizar uma operação, o centro de trabalho só estará disponível após o

processamento desta. Logo, o valor de Nop em média, desloca-se em unidades operacionais \bar{p} , que é o tempo médio de processamento das operações. De posse disto, é possível observar, conforme figura 6.28, que valores de k pequenos, tenham um incremento pequeno, enquanto que $k = 0.1$ torna a relação levemente exponencial. Valores de $k = 0.5$ praticamente conduzem a análise de que se houver muito tempo até a data de entrega o ATC torna-se zero, e começa a aumentar a prioridade em um horizonte de aproximadamente 5 operações. Para o valor de $k = 1$ este horizonte cai para 3, com $k = 2$ torna-se 2 e com $k = 3$, apenas preocupa-se com a operação que está na iminência de atrasar. A mesma análise pode ser feita a partir do momento que a operação está atrasada. A relação agora se torna inversa. Os fatores que praticamente não se preocupavam com uma folga grande, agora quando atrasam, tornam-se “estudantes” (fazendo uma alusão ao hábito de deixar tudo para a última hora), isto é, quando se atrasa a prioridade cresce rapidamente.

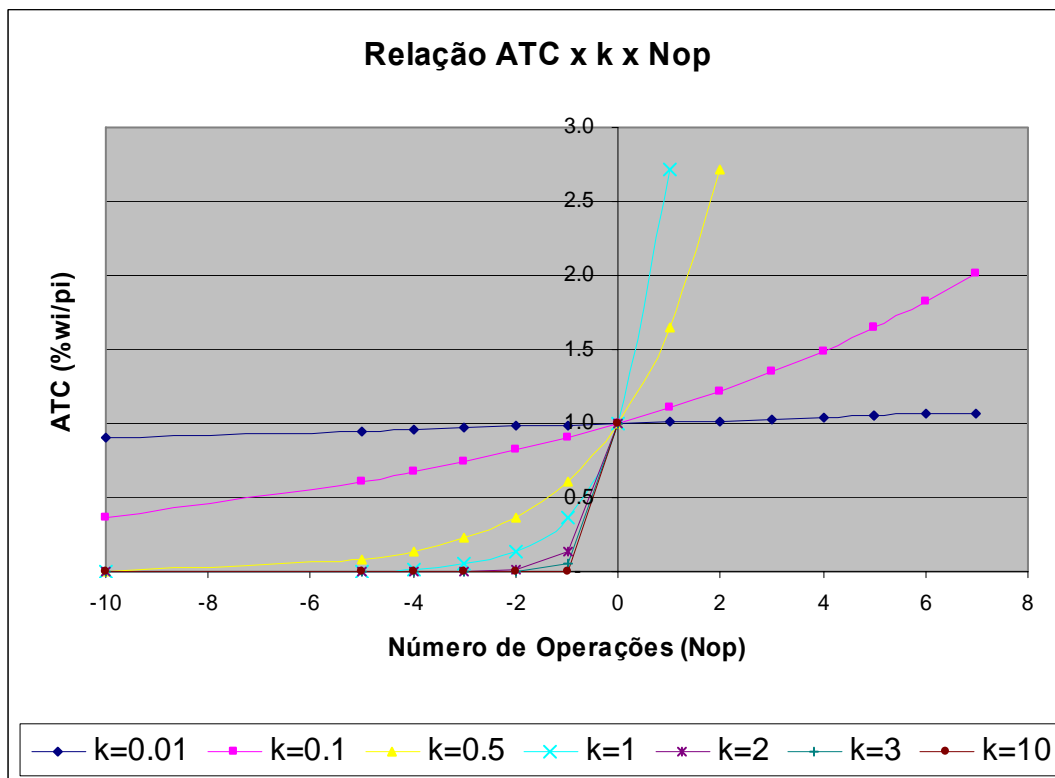


Figura 6.28 Relação do ATC em função de k e número de operações

Restrição de Data Mínima ou Data Mais Cedo

A última modificação na heurística, é com referência a data mínima ou data mais cedo. Uma operação não pode começar antes desta data, conforme explanado em tópico anterior. Para inserir esta alteração foi adicionado uma penalização exponencial, que

segue os mesmos padrões da fórmula (6.14), de tal sorte que a referência passa a ser a data mais cedo, conforme (6.15).

$$ATC = \frac{w_i}{p_i} \exp\{-k \cdot \overline{Nop}\} \cdot \exp\left\{-k_1 \cdot \left(\frac{t - r_i}{\overline{p}}\right)^+\right\} \quad (6.15)$$

Novamente, alterando a nomenclatura para tornar-se mais perceptível estes valores, fez-se a ociosidade igual a $(t - r_i)^+$, e da mesma forma que a análise feita anteriormente, quando divide-se por \overline{p} , encontra-se o número médio de operações ociosas que serão inseridas até a data o instante de processamento t , atingir a data mais cedo da operação. Desta forma a expressão fica:

$$ATC = \frac{w_i}{p_i} \exp\{-k \cdot \overline{Nop}\} \cdot \exp\{-k_1 \cdot \overline{Noc}^+\} \quad (6.16)$$

Observa-se aqui que se utilizam apenas valores positivos para Noc , isto porque a data mais cedo ou data mínima é uma restrição e não um objetivo. Logo, a partir do instante que a restrição é satisfeita, esta penalização é eliminada da equação.

Se considerarmos constante o valor $\exp\{-k \cdot \overline{Nop}\}$, a expressão resultante tem o mesmo formato da (6.13), portanto a mesma análise é pertinente. Neste caso, agora não mais se interessa permitir que uma operação seja sequenciada sem que esteja disponível, portanto aqui o interessante é um valor para $k_2 > 10$.

Exemplo 6.2 Aplicação da heurística ATC modificada

A tabela 6.9 exhibe um conjunto de 10 operações, com datas mínimas (r_i) variando uniformemente no intervalo $[0,25]$, os tempos de processamento (p_i) variando uniformemente no intervalo $[1,10]$, a data de entrega (di) variando entre $[10,30]$ e a prioridade (w_i) entre $[1,100]$, todos uniformemente distribuídos dentro do intervalo.

Tabela 6.9 Dados para o exemplo 6.2

Oper	ri	pi	di	wi
1	1	2	10	16
2	6	4	11	63
3	5	1	15	80
4	11	5	23	97
5	7	6	24	49
6	18	7	27	25
7	9	5	28	19
8	21	2	30	43
9	21	4	36	73
10	18	10	49	83

A figura 6.29 exibe um esboço das folgas das operações, de forma que no eixo x tem-se o tempo e no eixo y a operação. O valor esquerdo de cada operação corresponde a data mais cedo (*ri*) e o valor mais a direita corresponde a data prometida (*di*). Por exemplo, a operação possui seu valor mais a esquerda em 6 e o valor mais a direita em 11. Isto implica em uma folga de 6 unidade (5 da diferença mais um da própria data mínima que permite processamento).

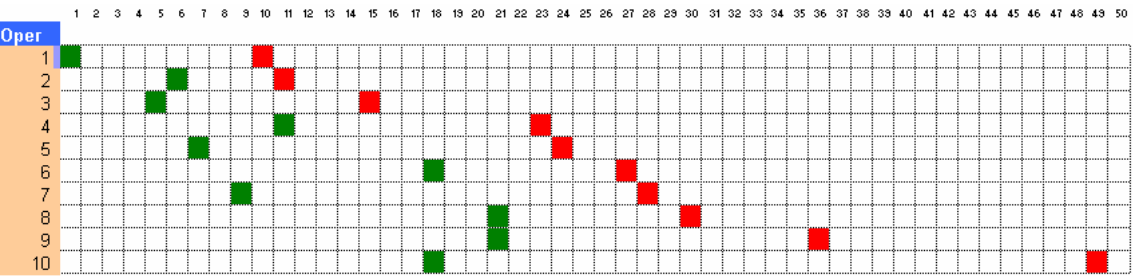


Figura 6.29 Esboço das folgas para as operações da tabela 6.9

O algoritmo consiste em calcular o *ATC*, utilizando-se a equação (6.16), para cada operação e selecionar o maior valor. A tabela 6.10 resume estes cálculos.

Tabela 6.10 Cálculo do *ATC* para as operações do exemplo 6.2, utilizando-se $k_1 = 0.5$ e $k_2 = 3$, $t = 0$.

Oper	rj	Tj	dj	wj	ATC	Seq
1	1	2	10	16	3.74	1
2	6	4	11	63	0.31	
3	5	1	15	80	1.43	
4	11	5	23	97	0.00	
5	7	6	24	49	0.03	
6	18	7	27	25	0.00	
7	9	5	28	19	0.00	
8	21	2	30	43	0.00	
9	21	4	36	73	0.00	
10	18	10	49	83	0.00	
\overline{p}		4.6			3.74	

O maior valor calculado corresponde à operação 1, valor 3.74. Isto é justificável pois é a única operação disponível para processamento no instante $t = 0$. Agora t é atualizado para $t = 2$ e o ATC é calculado novamente. A tabela 6.11 (a) exibe o cálculo para a segunda iteração. O maior valor calculado é 3.72, pode-se inferir que seu valor foi devido a, novamente ser a única novamente disponível para processamento. Da mesma forma na tabela 6.11 (b) e tabela 6.11 (c) o cálculo do ATC conduz a seleção das operações liberadas. A partir da tabela 6.11 (d) já existem mais de uma operação permitidas para o processamento e, portanto o parte referente à exponencial que diminui a prioridade das operações não liberadas não é mais considerada. O fator determinante para escolha agora dependerá da folga e por fim da prioridade por tempo de processamento. Sendo os cálculos efetuados pela fórmula do ATC . É exposto na tabela 6.11 todos os cálculos até a última operação ser sequenciada

Tabela 6.11 Iterações para o sequenciamento das operações do exemplo 6.2

(a)			(b)			(c)		
Oper	ATC	Seq	Oper	ATC	Seq	Oper	ATC	Seq
1	-	1	1	-	1	1	-	1
2	0.81		2	7.48	3	2	-	3
3	3.72	2	3	-	2	3	-	2
4	0.02		4	0.20		4	2.95	
5	0.07		5	0.80		5	3.64	4
6	0.00		6	0.00		6	0.01	
7	0.01		7	0.08		7	1.08	
8	0.00		8	0.00		8	0.01	
9	0.00		9	0.00		9	0.00	
10	0.00		10	0.00		10	0.00	
\bar{p}	4.89		\bar{p}	5.38		\bar{p}	5.57	
T	2		t	5		t	9	
(d)			(e)			(f)		
Oper	ATC	Seq	Oper	ATC	Seq	Oper	ATC	Seq
1	-	1	1	-	1	1	-	1
2	-	3	2	-	3	2	-	3
3	-	2	3	-	2	3	-	2
4	14.77	5	4	-	5	4	-	5
5	-	4	5	-	4	5	-	4
6	0.44		6	3.57		6	4.17	
7	1.84		7	2.91		7	3.52	
8	0.25		8	6.16	6	8	-	6
9	0.15		9	3.66		9	8.46	7
10	0.18		10	1.52		10	2.24	
\bar{p}	5.5		\bar{p}	5.6		\bar{p}	6.5	
T	15		t	20		t	22	

(g)				(h)		
Oper	ATC	Seq		Oper	ATC	Seq
1	-	1		1	-	1
2	-	3		2	-	3
3	-	2		3	-	2
4	-	5		4	-	5
5	-	4		5	-	4
6	5.38	8		6	-	8
7	4.66			7	7.40	9
8	-	6		8	-	6
9	-	7		9	-	7
10	3.42			10	5.56	10
\bar{p}	7.33			\bar{p}	7.5	
T	26			t	33	

A figura 6.30 exibe o gantt de operações após aplicação da heurística *ATC* modificada. O resultado final foi o atraso de duas operações, 6 e 7. Com um custo da função objetivo de $(33 - 27) * 25 + (38 - 28) * 19 = 340$.

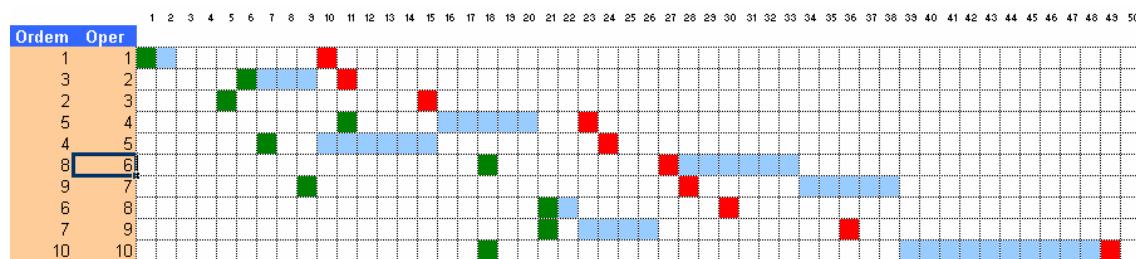


Figura 6.30 Gantt das operações após aplicações da heurística *ATC* modificada

6.4.5 Agendamento

Depois de concluída a fase de sequenciamento, a ordem em que as tarefas devem ser processadas estão definidas, porém ainda não se tem definido o exato instante em que cada tarefa deve iniciar.

Neste momento, o objetivo é apenas fornecer a real data de início em que uma operação deve iniciar. Nesta fase não há mais nada a fazer, a não ser de calcular a data, respeitando o calendário de trabalho.

Considere o exemplo 6.2 e a sequência fornecida pela tabela 6.11. O agendamento tem o objetivo de fornecer uma data para a operação. Suponha-se que a data de referência para início dos cálculos seja 01/01/2005 a 0:00 hora. Suponha-se ainda que o calendário permita trabalhar de segunda à sexta, das 8:00 às 12:00 e das

14:00 às 18:00. Desta forma as datas de início das operações são calculadas seguindo a sequência já fornecida. Estas datas são exibidas pela tabela 6.12 a seguir.

O agendamento não exige nenhum algoritmo de otimização, apenas deve ser levado em consideração os turnos e calendários de trabalho para cada recurso. A data de início é a maior data entre a data de término da última operação e a data mínima da operação em questão. Isto pode ser observado por exemplo, para operação 2, a qual poderia iniciar 01/01/2005 11:00, porém devido a data mínima, esta teve início somente em 01/01/2005 15:00. Outra questão a observar é que para o algoritmo de sequenciamento o tempo flui continuamente, o que na hora do agendamento não é verdade, pois, deve ser considerado o calendário. Assim, por exemplo, a operação 2, sequencia 3, inicia em 01/01/2005 16:00 e é executada por 4 horas, terminando dia 02/01/2005 10:00. Observa-se que a diferença real no calendário é de 18 horas.

Resumindo, esta fase se faz necessária somente por questões de visualização de dados e não modifica os resultados obtidos pelo sequenciador.

Tabela 6.12 Datas de início das operações

Sequencia	Operação	Tempo de Processamento (Horas)	Data Mínima	Data Início	Data Término
1	1	2	01/01/2005 09:00	01/01/2005 09:00	01/01/2005 11:00
2	3	1	01/01/2005 15:00	01/01/2005 15:00	01/01/2005 16:00
3	2	4	01/01/2005 16:00	01/01/2005 16:00	02/01/2005 10:00
4	5	6	01/01/2005 17:00	02/01/2005 10:00	02/01/2005 18:00
5	4	5	02/01/2005 11:00	02/01/2005 18:00	03/01/2005 15:00
6	7	5	02/01/2005 09:00	03/01/2005 15:00	04/01/2005 10:00
7	6	7	03/01/2005 10:00	04/01/2005 10:00	05/01/2005 09:00
8	8	2	03/01/2005 15:00	05/01/2005 09:00	05/01/2005 11:00
9	9	4	03/01/2005 15:00	05/01/2005 11:00	05/01/2005 17:00
10	10	10	03/01/2005 10:00	05/01/2005 17:00	09/01/2005 09:00

7 Resultados e Conclusões

O problema de sequenciamento classificado conforme seção 6.2, o qual este trabalho se propõe a resolver, não possui um algoritmo, até o presente momento (argumentação baseada nas pesquisas realizadas) que o resolva de maneira ótima em tempo computacional aceitável (entenda-se aceitável, devido a dinâmica de mercado, inferior a 1 dia). Extensível estes comentários até para pequenas instâncias, por exemplo, 10 a 100 operações. Portanto, não é possível comparar o algoritmo como um todo, de forma que a análise aqui seguida refere-se a cada fase do processo de sequenciamento, a saber: i) redução do número de variáveis decisórias, ii) designação de ferramentas e centros de trabalho e iii) sequenciamento ou ordenação das operações.

Uma abordagem sugerida para trabalho futuro é comparar os resultados de alguns pacotes comerciais, com a metodologia aqui exposta, já que esta também está embutida dentro de uma ferramenta comercial. Para uma comparação das técnicas, sugere-se (Zattar, 2004), que avaliou três sistemas *FCS – Finite Capacity System*, e analisou seu desempenho em relação aos indicadores globais da produção, tais como giro de estoque, redução de inventário, porém não os comparou em relação a qualidade de cada solução. Porém, não é possível efetuar a comparação da presente técnica com os três sistemas por ela analisados, por não haver disponível instâncias de problemas para verificação.

A seguir são analisadas as três fases do algoritmo, sendo que a primeira, redução do número de variáveis decisórias, e a segunda, designação de ferramentas e centro de trabalho, são enumeradas e explicitadas conclusões baseadas em critérios qualitativos, enquanto que a terceira, sequenciamento, os resultados são baseados em dados quantitativos.

7.1 Conclusões e Contribuições

O mérito do trabalho não está em resolver problemas específicos e compará-los com os métodos da literatura e sim a abordagem holística de um problema de planejamento que conduz a programação da produção de uma maneira mais robusta. Visa-se não só uma técnica em particular, como o sequenciamento ou designação de centros de trabalho, ou teoria das restrições, mas sim uma abordagem sistêmica.

O conjunto de técnicas juntas é que sinergicamente conduzem a uma boa programação da produção. A principal técnica, objeto de estudo mais detalhado neste trabalho é a heurística modificada para sequenciamento. Mas esta sozinha, não permitiria a solução do problema como um todo, necessitando de amparo de outras ferramentas ou teorias.

Portanto para a solução do problema de sequenciamento, foi utilizado um conjunto de técnicas que visaram realizar o planejamento finito das necessidades empresariais. Este planejamento finito consistiu em levar em conta as capacidades dos recursos, e, além disto, sequenciar operações. Estas técnicas para efetuar este sequenciamento podem ser resumidas abaixo:

- ☑ **Teoria das Restrições**: Utilizada para reduzir o número de variáveis de decisão. Além disto, serviu como balizador ao fornecer as datas mais cedo e mais tarde das tarefas processadas pela restrição. As datas de inícios das operações não restritivas são dadas em função das operações restritivas. Para prover segurança devido a não se levar em conta estas operações, utilizam-se pulmões (conforme visto em 4.5.2), que servem como “sorvedouros” das flutuações estatísticas.
- ☑ **Programação Dinâmica**: utilizada para realizar, tanto as designações de ferramental quanto as do centro de trabalho.
- ☑ **Heurística ATC Modificada - Regra de Despacho Dinâmica**: Utilizada para prover a ordem das operações restritivas.

Cada uma destas técnicas utilizadas proporcionou vantagens e desvantagens para o modelo completo. Na seqüência destacam-se estas vantagens e desvantagens.

7.1.1 Teoria das Restrições

Um fato exposto em (Zattar, 2004) informa que ao se utilizar a *TOC*, obtém-se os seguintes benefícios:

- ☑ Redução nos prazos de entrega;
- ☑ Elevação do nível de serviço à clientes;
- ☑ Aumento no ganhos pela otimização das restrições;
- ☑ Redução dos estoques;
- ☑ Redução das despesas operacionais

Porém, para o escopo deste trabalho, a *TOC* alicerçou a redução do número de variáveis decisórias e simplificou o problema, justificando a aplicação de algoritmos

mais simples e com ordem de complexidade pseudo-polinomiais (balanceamento das cargas de ferramental e centros de trabalho) e quadráticos (sequenciamento), para solução dos problemas.

Porém o preço pago não se reflete no sequenciamento e sim nos efeitos colaterais, ie, o planejamento dos demais centros não restritivos. Esta reação adversa aparece, principalmente quando as capacidades dos GM's estão muito próximas entre si. Como a *TOC* supõe que o recurso que impõe o fluxo produtivo é exclusivamente o recurso restritivo, parte-se do princípio que os demais recursos tem “capacidade infinita”. Isto implica em ser possível programar mais de uma operação ao mesmo instante no mesmo centro de trabalho, e conseqüentemente atrasos ou antecipações ocorrerão.

Para justificar esta simplificação a *TOC* propõe inserir em alguns pontos e principalmente antes da restrição um pulmão de tempo. Este pulmão tem o objetivo de absorver as flutuações estatísticas do processo, e na visão deste trabalho, ignorar a capacidade dos demais recursos e transferir o “erro” de sua programação para este pulmão. Implicando em antecipar as operações para evitar que atrasem o recurso restritivo.

Sugere-se como trabalho futuro, propor técnicas para dimensionamento do pulmão, de forma a contemplar, além das flutuações estatísticas, os erros de programação.

7.1.2 Programação Dinâmica

A utilização de Programação Dinâmica nem sempre conduz a resultados ótimos, como exemplo (Silveira e Morabito, 2002) que descrevem uma técnica de programação dinâmica para um problema de corte bidimensional guilhotinado que não garante o ótimo porém fornece, se não o ótimo, um limitante muito próximo permitindo sua estimativa.

Motivado por estes resultados, foram desenvolvidos os algoritmos de designação de ferramental e centros de trabalho. O primeiro atuando de forma independente e o segundo recebendo as informações do primeiro, que serviu como poda do espaço de alternativas válidas (um centro só era válido se a uma operação antecessora que usasse o mesmo ferramental estivesse designada no mesmo centro). As conclusões referentes a este algoritmo são extraídas de (Darú, 2005).

Complexidade

O algoritmo apresentado neste trabalho possui uma complexidade $O(t.f^2)$, onde t é o número de operações e f é o número de ferramentas, enquanto que uma heurística gulosa possui complexidade $O(t.f)$. Isto implica em tempos pseudo-polinomiais, os quais não tendem a possuir diferenças de performance significativas, com o aumento do número de variáveis.

Espaço de busca

Uma vantagem observada pelo algoritmo proposto é pela sua visão em “profundidade” dada pela avaliação de alternativas passadas, implicando em obter soluções melhores. Isto é devido exatamente pelo preço pago em sua complexidade.

Princípio de Bellman

Uma questão identificada após o desenvolvimento do algoritmo foi a violação do princípio de Bellman, o qual implica na não obtenção da resposta ótima. Isto implicou que a resposta é dependente da sequência dos estágios escolhidos, ou seja da ordem em que as tarefas são posicionadas.

Qualidade da Solução

Devido ao objetivo não ser linear, a função obtém bons resultados e comparando-se com a uma heurística gulosa (ver Daru, 2005), percebe-se que se dá de maneira mais eficiente, como pode ser visto nas soluções desta heurística.

O critério adotado que mostrou dar bons resultados foi a ordenação das tarefas por data mais cedo, objetivando alocar as ordens conforme se tornassem disponíveis e um segundo critério de desempate por data mais tarde, para respeitar os prazos de entrega.

O ponto negativo da heurística é a relativa complexidade para se inserir alterações, isto foi verificado, posteriormente quando do tratamento da designação dos centros por prioridade, ie, uma operação deve ser alocada no centro prioritário, a não ser que (a) o centro não possua mais disponibilidade ou (b) a operação atrasará.

7.1.3 Heurística ATC modificada

Para amparar a avaliação da heurística modificada, é gerado um conjunto de amostras com o objetivo de simular dois cenários produtivos: (a) um cenário com folga ou tempo suficiente para processar as operações e (b) um cenário apertado com prazos curtos e com inevitáveis atrasos. Para efetuar a geração das amostras seguiu-se a abordagem realizada por (Valente, 2003), com algumas adaptações.

As amostras geradas são dependentes de dois fatores, chamados de *TF* ou *Tardiness Factor* e *RDD* ou *Range Due Date*. O primeiro tem uma variação no intervalo $[0,1]$ e se refere a possibilidades de ocorrerem atrasos. Quando seu valor é próximo de 1, significa que as datas prometidas estão concentradas no início do horizonte de planejamento, implicando em atrasos. O inverso, um valor próximo de 0 conduz a análise de que as datas são para o fim do horizonte, implicando em poucos atrasos. O valor *RDD*, mede a variabilidade das datas prometidas, sendo parecido com o desvio padrão da estatística. Isto conduz a análise que em um conjunto de amostras com baixa variabilidade, somente *TF* é relevante, enquanto valores altos de *RDD* não permitem uma análise direta. As fórmulas (7.1) e (7.2) exibem sua definição.

$$TF = 1 - \frac{\bar{d}}{C_{\max}} \quad 7.1$$

$$RDD = \frac{d_{\max} - d_{\min}}{C_{\max}} \quad 7.2$$

Onde,

\bar{d} = média das datas prometidas;

d_{\max} = maior data prometida;

d_{\min} = menor data prometida;

C_{\max} = maior data de entrega.

Como a técnica sugerida por (Valente, 2003) não trata das datas mínimas, aqui é sugerido a seguinte alteração. Sabendo-se que \bar{d} é $\frac{\sum d_i}{n}$ e substituindo-se a data prometida, d_i por tempo de processamento adicionado da data mínima e da folga, $p_i + r_i + f_i$, obtém-se $\frac{\sum (p_i + r_i + f_i)}{n}$. Para o caso de ao primeiro instante todas as operações estiverem disponíveis, tem-se $r_i = 0$, retornando-se ao caso original. Definindo-se o fator de folga (*FF*), conforme (7.3), pode-se encontrar uma relação entre *TF* e *FF* dada pela equação (7.4).

$$FF = 1 - \frac{\bar{f}}{C_{\max}} \quad 7.3$$

$$TF = FF - \frac{\bar{r} + \bar{p}}{C_{\max}} \quad 7.4$$

Onde

\bar{f} = folga média, onde folga é dada por $di-pi-ri$;

di = data prometida para operação i ;

pi = tempo de processamento da operação i ;

ri = data de liberação ou mínima da operação i ;

\bar{r} = data de liberação média;

\bar{p} = tempo de processamento médio das operações.

A segunda medida, RDD , tem o intuito de medir a variabilidade do fator TF , fornecendo uma espécie de desvio padrão. Como o RDD está relacionado com a data prometida, e sabendo que a data prometida pode ser definida em função da data de liberação - ri , tempo de processamento - pi e da folga - fi , através da equação $di = ri + pi + fi$, pode-se estimar a variância da data prometida - $Var(DD)$ como sendo a variância da soma das três variáveis. Desta forma $Var(DD) = Var(r+p+f)$. Supondo que as variáveis d , r e p são independentes entre si, vem que $Var(DD) = Var(r)+Var(p)+Var(f)$. Como RDD é uma estimativa de $Var(DD)$, então $RDD = RF + RP + RR$. Sendo RF o range da folga - $(f_{max} - f_{min})/C_{max}$, RP como o range do tempo de processamento $(p_{max} - p_{min})/C_{max}$ e RR o range da data mínima - $(r_{max} - r_{min})/C_{max}$. De forma que todos são estimativas, respectivamente, de $Var(r)$, $Var(p)$ e $Var(f)$. A expressão (7.5) mostra esta relação.

$$RDD = \frac{(f_{max} - f_{min}) + (r_{max} - r_{min}) + (p_{max} - p_{min})}{C_{max}} \quad 7.5$$

Com isto posto, é gerado um conjunto de amostras com distribuição uniforme para cada variável. O número de operações para cada amostra foi escolhido de tal forma que fosse possível encontrar o valor ótimo da amostra utilizando-se uma formulação em programação linear inteira - PLI , conforme código no Anexo A, em um tempo computacional aceitável por simulação. Este valor foi definido em 10, o que conduz a um tempo de aproximadamente 1 minuto por amostra.

O valor de C_{max} foi escolhido para ser aproximadamente uma semana produtiva em horas, sendo escolhido um valor 50. A variação do tempo de processamento foi escolhida para que fosse possível processar todas as operações dentro do intervalo de C_{max} , ou seja o intervalo $[1,9]$, com uma média de 5 horas por operação. O valor das datas de liberação foram escolhidos dentro do intervalo $[0,25]$ uniformemente

distribuídos, supondo que as operações terão o material necessário para o seu processamento dentro da metade do horizonte de planejamento C_{max} .

O valor do parâmetro k_2 é definido como 10, devido ao não desejo de permitir uma operação ser sequenciada antes de atingir a data mínima. É possível estimar este valor a partir da figura 6.28. O valor de k_1 foi simulado no intervalo $[0.1, 2.9]$ e escolhido o valor que obteve o melhor resultado para cada amostra. O intervalo das folgas é dependente de TF e RDD . A distribuição de TF é obtida substituindo-se os valores de \bar{p} - (12.5, proveniente da média do intervalo $[0,25]$) e \bar{r} (5, proveniente da média do intervalo $[1,9]$) em (7.4); isto conduz a $TF = FF - 0.35$. Da mesma forma, tem-se $RDD = RF + 0.66$. Atribuindo-se para TF os valores $\{0.2, 0.5\}$, para simular um cenário apertado e outro um pouco mais relaxado, e para RDD o valor 0.8, obtém-se as distribuições das variáveis conforme tabela 7.1 (a) e (b).

Tabela 7.1 Parâmetros e intervalos das distribuições das variáveis para simulação

(a)		(b)	
r_i	$[0,25]$	r_i	$[0,25]$
TF	0.2	TF	0.5
FF	0.55	FF	0.85
\bar{f}	23	\bar{f}	7
RDD	0.8	RDD	0.8
RF	0.14	RF	0.14
f_i	$[15,30]$	f_i	$[0,15]$

$$C_{max} = 50, n = 10, p_i \in [1,9] \text{ e } w_i \in [1,10]$$

Para ilustrar cada ambiente, exibe-se na figura 7.1 e na figura 7.2 uma amostra de cada cenário. É possível perceber que a primeira figura possui um ambiente que implicará em menos tarefas atrasadas, devido as folgas serem maiores e consequentemente datas de entrega também maiores, enquanto que o segundo, possui datas de entrega mais cedo, implicando em um cenário com inevitáveis atrasos.

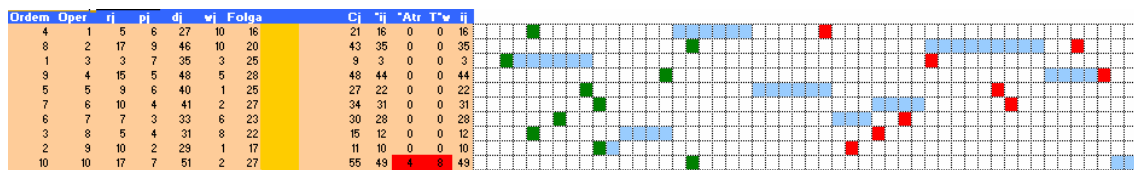


Figura 7.1 Amostra utilizando-se $TF = 0.2$

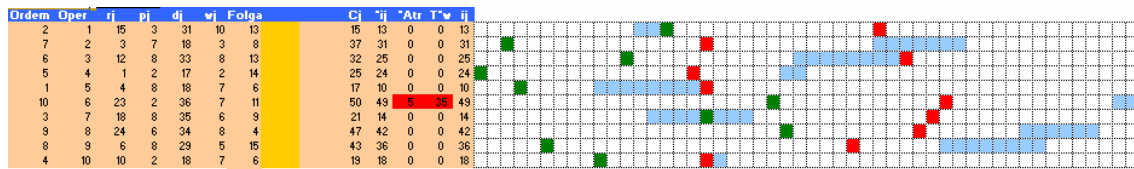


Figura 7.2 Amostra utilizando-se $TF = 0.5$

Para cada cenário gerou-se 30 amostras e calculou-se o valor ótimo da função objetivo e o valor obtido aplicando-se a heurística ATC modificada. A tabela 7.2 exibe os resultados para $TF = 0.2$ e a tabela 7.3 para $TF = 0.5$. As tabelas também exibem o valor de k_l que deu o melhor resultado para instância, bem como a distância que a regra ficou do ótimo dada por $(ATC - \acute{O}timo)/ATC$ e o tempo necessário para calcular o valor ótimo via *PLI*, utilizando-se o software Lingo® e a modelagem conforme Anexo A.

Tabela 7.2 Resultados obtidos para $TF = 0.2$

Instância	k	Ótimo	ATC	%	Tempo
1	0.2	0	0	0.0%	00:00:01
2	0.6	0	0	0.0%	00:00:01
3	0.1	0	0	0.0%	00:00:01
4	0.1	0	0	0.0%	00:00:01
5	0.1	0	0	0.0%	00:00:02
6	0.9	0	0	0.0%	00:00:30
7	0.1	0	0	0.0%	00:00:43
8	0.3	0	0	0.0%	00:00:51
9	0.4	0	0	0.0%	00:01:18
10	1.0	0	0	0.0%	00:01:19
11	0.9	14	14	0.0%	00:02:43
12	0.7	0	0	0.0%	00:03:06
13	0.4	10	10	0.0%	00:05:01
14	1.3	4	4	0.0%	00:05:30
15	0.8	9	9	0.0%	00:07:39
16	1.3	25	25	0.0%	00:08:47
17	0.1	42	42	0.0%	00:09:33
18	1.2	39	39	0.0%	00:11:37
19	1.2	28	28	0.0%	00:14:03
20	0.9	74	74	0.0%	00:14:08
21	0.6	63	63	0.0%	00:26:45
22	0.7	14	14	0.0%	00:29:32
23	0.4	117	132	11.4%	00:06:41
24	0.3	83	95	12.6%	00:13:20
25	0.8	122	143	14.7%	01:13:00
26	0.3	77	94	18.1%	00:11:29
27	0.5	56	77	27.3%	01:06:29
28	0.4	16	24	33.3%	00:11:30
29	0.7	43	67	35.8%	00:08:46
30	0.1	8	16	50.0%	00:10:58
Média	0.6			6.8%	00:11:31
desvpad	0.4			13.3%	00:17:30

Para a tabela 7.2 o algoritmo encontrou a melhor solução em 73% das vezes o que implica que para ambientes relativamente folgados o algoritmo apresenta um bom desempenho. Outro fator relevante é o tempo de processamento médio para cada instância na procura do ótimo, que ficou na casa dos 10 minutos. Isto exhibe que a busca pelo ótimo não é possível para ambientes reais de produção, sendo necessário o uso de heurísticas.

Tabela 7.3 Resultados obtidos para $TF = 0.5$

Instância	k	Ótimo	ATC	%	Tempo
1	0.3	72	72	0.0%	00:00:42
2	1.2	50	50	0.0%	00:00:09
3	0.1	105	105	0.0%	00:00:36
4	0.8	7	7	0.0%	00:00:01
5	1.0	34	34	0.0%	00:00:04
6	0.3	41	41	0.0%	00:00:47
7	0.1	111	111	0.0%	00:00:50
8	0.1	193	193	0.0%	00:01:01
9	0.1	614	622	1.3%	00:42:26
10	0.6	332	345	3.8%	00:09:02
11	0.2	223	233	4.3%	00:04:31
12	0.3	141	148	4.7%	00:01:03
13	0.5	154	164	6.1%	00:01:38
14	0.1	265	287	7.7%	00:10:00
15	0.7	117	128	8.6%	00:03:36
16	0.5	296	325	8.9%	00:06:10
17	0.3	156	174	10.3%	00:00:33
18	0.6	133	149	10.7%	00:00:41
19	0.3	92	105	12.4%	00:01:18
20	0.3	206	239	13.8%	00:03:25
21	0.1	386	448	13.8%	00:08:41
22	0.6	80	94	14.9%	00:01:06
23	0.6	74	89	16.9%	00:01:48
24	0.2	192	232	17.2%	00:02:19
25	0.2	109	153	28.8%	00:00:47
26	0.1	109	156	30.1%	00:01:50
27	0.7	59	90	34.4%	00:00:03
28	0.1	98	163	39.9%	00:02:40
29	0.1	299	536	44.2%	00:00:54
30	0.9	37	84	56.0%	00:00:15
Média	0.4			13.0%	00:03:38
desvpad	0.3			14.9%	00:07:50

O algoritmo encontrou a solução ótima em 27% das simulações e ficou a 5% do ótimo em 40% das simulações. Levando em consideração que o algoritmo deve ser ao mesmo tempo *performático* e eficiente na otimização, os objetivos atingidos foram

satisfatórios. Nas 30 instâncias a média ficou em 13% do ótimo. Levando em consideração que o ótimo levou em média 3 minutos para ser obtido para uma instância de 10 operações, e a regra possui um tempo inferior a 1 segundo, os resultados são desejáveis sob o enfoque produtivo. Lembrando que em um ambiente real de produção as operações chegam facilmente a 4000, e que alterações ocorrem quase que a todo instante devido as flutuações estatísticas.

Outro comentário interessante é que o valor de k_1 próximo de 0.5 é que obteve na média os melhores resultados, tanto para ambientes com TF alto quanto baixo. Porém há que se destacar que há uma grande incidência de valores altos para k_1 em algumas simulações, sugerindo que para empresas com TF baixo, sejam utilizados valores levemente superiores a 0.5.

As simulações foram realizadas, objetivando-se minimizar o atraso das operações prioritárias, porém nem sempre no chão de fábrica isto é o desejável. Em algumas situações o usuário deseja respeitar a data de entrega, em outras a prioridade. Antecedências nem sempre são bem vindas, devido ao custo de carregamento de inventário, aumento do estoque em processo, entre outros.

Devido as constantes alterações no cenário produtivo é proposto no Anexo B uma externalização dos parâmetros. Esta externalização permite ao algoritmo transformar-se em uma regra de despacho híbrida ou fixa por um critério como data de entrega, prioridade, tempo de processamento ou folga, ou ainda um híbrido entre eles.

Os maiores problemas do algoritmo encontram-se quando as variáveis decisórias relacionadas ao objetivo começam a conflitar. Lembrando que em situações práticas as variáveis importantes são (i) a prioridade, (ii) a data de entrega e (iii) a data mais cedo, e em segundo plano (iv) o tempo de processamento. O algoritmo pode trocar a seqüência das operações devido aos parâmetros utilizados como k_1 ou k_2 e não devido as reais necessidades do usuário.

7.2 Sugestões para Trabalhos Futuros

Podem-se distinguir dois tipos de sugestões, o primeiro grupo refere-se ao aprimoramento ou incorporação de novas funcionalidades a heurística proposta enquanto que o segundo refere-se ao caráter exploratório dos resultados aqui obtidos.

7.2.1 Sugestões de Aprimoramento

Conforme visto no capítulo que trata de sequenciamento, existem diversos fatores que podem ser levados em conta quando da realização do sequenciamento de tarefas, alguns fatores relevantes na prática não foram levados em consideração neste trabalho e são enumerados a seguir:

1. **Setup.** A questão do setup é altamente relevante para grande parte das indústrias, principalmente as que trabalham com produtos com características relevantes para o processo de fabricação, como cor ou numeração (exemplo de indústrias calçadistas e tecelagem). Uma sugestão é utilizar a heurística proposta por (Lee et al, 1997).
2. **Lote.** Definir o tamanho do lote de fabricação é outra questão muito utilizada no ambiente fabril. Este determina o tempo mínimo que uma máquina deve processar um determinado item para valer a pena sua fabricação. Isto poderia ser realizado utilizando-se uma técnica de agrupamento dentro de um horizonte pré-determinado.
3. **Rede Pert.** Neste trabalho a rede pert foi considerada como definida durante a fase de explosão, entrando no algoritmo como datas mais cedo de fabricação. Mas para os casos onde há recirculação no recurso restritivo, isto é, uma operação passa duas vezes pelo gargalo, porém um deve passar antes que a outra, não há um tratamento direto, a não ser como já exposto pelas fase de explosão que impõe datas mais cedo, e mais tarde.

7.2.2 Sugestões de Comparação

Estas sugestões visam comparar a heurística, com um enfoque avaliativo, em relação a outras técnicas de sequenciamento.

1. **CSP.** Utilizar a técnica de satisfação de restrições para realizar o sequenciamento. Esta técnica permite encontrar uma solução satisfatória ao problema em tempo computacional aceitável.
2. **Aplicações de Regras de Dominância.** Implementar as regras de dominância sugeridas em (Akturk e Yildirim, 1999) e levantar os impactos na melhoria da qualidade da solução.

3. **Princípio de Bellmann.** Identificar quais os impactos da ordenação das variáveis que correspondem aos estágios, nos algoritmos de designação de ferramental e centros de trabalho, referentes à busca do ótimo, no resultado final, bem como levantar qual o melhor critério a ser utilizado.

8 Referências Bibliográficas

- (Abepro, 2005) Internet: www.abepro.org.br. Último acesso 04/04/2005.
- (Akturk e Yildirim, 1999) Akturk, M. Sellim e Yildirim, M. Bayram. *A New Dominance Rule for the Total Weighted Tardiness Problem*. Production Planning & control, 1999, Vol 10, no. 2, 138-149.
- (Baker, 1974) Baker, Kenneth R. *Introduction to Sequencing and Scheduling*. Ed. John Wiley e Sons, 1974.
- (Bellmann, 1974) Bellman, R. E. Dynamic Programming, Princeton University Press, 1974.
- (Cook, 1971) Cook, Stephen A. *The Complexity of Theorem-Proving Procedures*. Universidade de Toronto, 1971.
- (Cook, 2000) _____. *The P versus NP Problem*. Universidade de Toronto, 2000.
- (Cook, 2002) _____. *The Importance of the P versus NP Question*. Universidade de Toronto, 2002.
- (Cox III e Blackstone, 1998) Cox III, James F e Blackstone Jr., John H. *APICS Dictionary*. 9a Ed. APICS, 1998.
- (Cox e Spencer, 2002) Cox, James F. e Spencer, Michael S. *Manual da Teoria das Restrições*. Bookman, 2002.
- (Darú, 2005) Darú, Gilsiley H e Lacerda, Vanessa C. *Utilização de Programação Dinâmica Multirotulada para Balanceamento do Uso de Ferramenta*. XXVIII CNMAC.
- (Davis et al, 2003) Davis, Mark M.; Aquilano,, Nicholas J. E Chase, Richard B. *Fundamentals of Operations Management*. Editora McGraw-Hill, 2003.
- (Devlin, 2004) Devlin, Keith. *Os Problemas do Milênio*. Editora Record, 2004.
- (Gaither et al, 1999) Gaither, Norman e Frazier, Greg. *Administração da Produção e Operações*. 8ª Edição, Editora Pioneira, 1999.
- (Goldratt e Fox, 1992) Goldratt E. M. *A Meta*. Ed. Educator, 1992.
- (Guerreiro, 1999) Guerreiro, Reinaldo. *A Meta da Empresa, Seu Alcance sem Mistérios*. Editora Atlas, 2ª Edição, 1999.
- (Jain, 1988) Jain, Anant Singh. *A Mult-level Tabu Search Framework for the Deterministic Job-Shop Problem*. Phd thesis, University of Dundee, 1988.

- (Lee et al, 1997) Lee, Y. H., Bhaskaran, K. E Pinedo M. *A heuristic to Minimize the Total Weighted Tardiness with Sequence-dependent Setups*. IIE Transactions (29), pp. 45-52 (1997).
- (Levin, 1973) Levin, L. Universal search problems. *Problemy Peredachi Informatsii*, 9(3):265-266, 1973.
- (Miyazawa, 2001) Miyazawa, Flávio K. *Programação Inteira*. Universidade de Campinas, 2001.
- (Pinedo, 2002) Pinedo, Michael. *Scheduling Theory. Algorithms and Systems*. Prentice Hall, 2a Edição.
- (Rachamadugu, 1982) Rachamadugu, Ram Mohan V. e Morton, Thomas F. *Myopic Heuristic for the Single Weighted Tardiness Problem*. Carnegie-Mellon University, 1982.
- (Slack e Nigel, 97) Slack, Nigel et al. *Administração da Produção*. Editora Atlas, 1997.
- (Silveira e Morabito, 2002) Silveira, Rejane e Morabito, Reinaldo, Um Método Heurístico Baseado Em Programação Dinâmica para o Problema de Corte Bidimensional Guilhotinado Restrito, rev. Gestão e Produção, v.9, n.1, p.78-92, abr. 2002.
- (Utku, 1999) Utku, D. Hakan. *Complexity of Single Machine Hierarchical Scheduling*. Universidade Bilkent, Ankara, 1999.
- (Valente, 2003) Valente, Jorge M. S. *Using Instance Statistics To Determine the LookAhead Parameter Value in the ATC Dispatch Rule, Making a good heuristic better*. Working Papers da FEP, trabalho em curso número 127, Faculdade de economia da Universidade do Porto, 2003, Portugal.
- (Zattar, 2004) Zattar, Izabel Cristina. *Análise da Aplicação dos Sistemas Baseados no Conceito de Capacidade Finita nos Diversos Níveis da Administração da Manufatura Através de Estudos de Caso*. Dissertação de Mestrado, Departamento de Engenharia Mecânica, UFSC, 2004.

Glossário

ABEPRO: Acrônimo de Associação Brasileira de Engenharia de Produção;

APICS: Acrônimo de Association for Production and Inventory Control System;

APO: Acrônimo para Administração da Produção e Operações;

CRP: Acrônimo de Capacity Resource Planning, que objetiva verificar a utilização da capacidade dos recursos;

DBR: Acrônimo de Drum, Buffer, Rope. Mesmo conceito de TPC;

ERP: Acrônimo de Enterprise Resource Planning. São sistemas integrados de informação, de estruturas abrangentes e complexas que tentam tratar, integradamente, o máximo do modelo de informação da organização.

Estoque de Segurança: é o nível mínimo de estoque planejado com o propósito de proteção contra as flutuações de demanda ou fornecedores;

FCS: Acrônimo de Finite Capacity System ou Planejamento Finito da Capacidade.

Gargalo: é algo que impede a empresa de ganhar mais dinheiro;

Grupo de Máquina: Uma área específica da produção, consistindo de uma ou mais máquinas com capacidades idênticas, que podem ser consideradas como uma unidade para propósitos do CRP e do sequenciamento;

Horizonte de Planejamento: Período a ser considerado para se realizar o planejamento mestre. Conforme Dicionário da APICS deve ser no mínimo tão longo quanto o tempo do item com maior tempo de produção. Consiste, geralmente, de espaços temporais de uma semana, estendendo-se até oito semanas;

IIIE: Acrônimo de International Institute of Industrial Engineering;

JIT: Acrônimo para Just In Time;

Lead Time: é o tempo necessário para que um pedido, lote, ou algo que se esteja produzindo, flua de uma etapa a outra;

Lote Fixo: é a quantidade mínima a ser produzida de forma a minimizar os custos produtivos (como tempos de preparação de máquina);

MPS: Acrônimo para Master Program Schedule;

MRP I: Acrônimo de Material Resource Planning, tem por objetivo realizar o planejamento das necessidades de materiais, tomando como base os pedidos, ordens e previsões menos o estoque;

MRP II: Acrônimo de Material Requirement Planning;

MTO: Acrônimo de Make To Order, ou fabricar sob encomenda, ou pedido;

MTS: Acrônimo de Make To Stock, ou fabricar para estoque;

OPT: Acrônimo de Optimized Production Technology;

PLI: Acrônimo de Programação Linear Inteira

RCCP: Rought-Cut Capacity Planning ou planejamento da capacidade de médio prazo;

ROI: Acrônimo de Return Over Investment ou Retorno sobre o Investimento.

RRC: Acrônimo de Recurso com Restrição de Capacidade, é sinônimo de algo produtivo que tem a menor capacidade de fabricação;

SFC: Acrônimo para Shop Floor Control;

SOP: Acrônimo de Sales Operation Planning ou plano de vendas corporativo;

Time Fence: Período de Congelamento;

TOC: Acrônimo de Theory of Constraints ou Teoria das Restrições;

TPC: Acrônimo de Tambor, Pulmão Corda. Processo de programação da produção, utilizado na TOC;

Anexo A – Código de Programação Linear Inteira Utilizado no software LINGO

```

MODEL:
SETS:
    Oper / @OLE( '.\Análise ATC.xls' ) / : rj, dj, tj, wj, ij, Atraso,
    Atrasou, Antecipacao;
    Sequencia( Oper, Oper );
    Ordem ; ! Representa o Numero de moldes Molde i usados na injetora
j;
    Tipo / 1 / : Objetivo;
ENDSETS
DATA:
rj = @OLE( '.\Análise ATC.xls' ) ;
dj = @OLE( '.\Análise ATC.xls' ) ;
tj = @OLE( '.\Análise ATC.xls' ) ;
wj = @OLE( '.\Análise ATC.xls' ) ;
@OLE( '.\Análise ATC.xls' ) = Ordem;
@OLE( '.\Análise ATC.xls' ) = ij;
@OLE( '.\Análise ATC.xls' ) = Atraso;
ENDDATA
!Objetivo Minimizar Atraso dos Pedidos Prioritários;
Min = @SUM(Oper(I) : Atraso * wj ) ;
!Restrições, cálculo do Atraso;
@FOR( Oper(I) :
    Atraso >= 0 ;
    Atraso >= ij + tj - dj - 1 ;
);

!Restrição de Sequencias;
@FOR( Oper(I) :
    @FOR( Oper(J) | J #GT# I :
        ij(I) + 100*ORDEM(I,J) >= ij(J) + tj(J) ;
        ij(J) + 100*(1-ORDEM(I,J)) >= ij(I) + tj(I) ;
    )
);

!Restrição de Datas Mínimas;
@FOR( Oper(I) : ij >= rj; );
@FOR( Sequencia : @BIN(Ordem) );
END

```

Anexo B – Externalização de Parâmetros e Transformação da Heurística ATC para outras Heurísticas

Por questões práticas, impor uma tecnologia ao usuário nem sempre é o melhor a fazer. Por melhor que seja a técnica desenvolvida, sempre terão desejos adversos, os quais são justificáveis por diversos motivos. Para se adequar e tornar flexível a formulação, inseriu-se variáveis “FLAG”, as quais tem a missão de ativar ou desativar, determinadas regras. A equação geral da heurística com estes flags, é exibida abaixo:

$$ATC = \frac{\left((Status^{F_1} \cdot k_1) + \left(1 - \frac{w^*}{w_{\max}^*} \right) \cdot k_2 \right)^{F_1 \vee F_2}}{p_i^{F_3}} \cdot \exp\left\{ -k_3^{F_4} \cdot \overline{Nop} \right\} \cdot \exp\left\{ -k_4 \cdot \overline{Noc}^+ \right\} \quad 1$$

Os flags F1 a F4 são booleanos, ou seja, podem assumir apenas os valores 0 ou 1.

Com isto, é possível gerar as demais heurísticas de despacho com determinadas combinações dos flags. Estas combinações são descritas na tabela a seguir.

Heurística	Valor			
	F1	F2	F3	F4
SJF	0	0	1	0
Prioridade	0 ou 1	1	0	0
EDD	0	0	0	1
WSJF	0 ou 1	1	1	0

Anexo C – Código Fonte das Rotinas em linguagem C, utilizadas para Sequenciamento

```

void dbapi004seq0(float *fTempoProcessamento,
                  float *fFerramentaOP )

{
    int iTempoProcessamentoCol;
    int iFerramentaCol;

    int iNumOP;
    int iNumCT;
    int iCTAtual;
    int iCTAnt;
    int opAntecessora, iOp;
    int iEstagio;

    float fNorma = 0;
    float fNormaAux = 0;
    float fNormaEuclides = 0;
    float fNormaEuclidesAux = 0;
    float fOp = 0;
    float fSolucao = 0;
    float fFerramenta = 0;
    float iTempoProcessamento = 0;
    float iTempoProcessamentoAnt = 0;
    float iDataMinima = 0;
    float fIP = 0;
    float *fEstagioAnt,
           *fEstagioAtual,
           *fCarga,
           *fTrajetoria,
           *fLabel;

    Union un;
    FILE *fp;
    int iAux,jAux;

    //busca o numero de operacoes(iNumOP) e centros de trabalho(iNumCT)
    un.f = fTempoProcessamento[0];
    iNumOP = un.i;
    un.f = fTempoProcessamento[1];
    iNumCT = un.i;

    un.f = fTempoProcessamento[1];
    iTempoProcessamentoCol = un.i;

    un.f = fFerramentaOP[1];
    iFerramentaCol = un.i;

    fEstagioAnt = (float*)malloc(sizeof(float) * iNumCT * iNumCT + 8);
    memset(fEstagioAnt,0, sizeof(float) * iNumCT * iNumCT + 8);
    fEstagioAnt[0] = fEstagioAnt[1] = (float)iNumCT;
    fEstagioAtual = (float*)malloc(sizeof(float) * iNumCT * iNumCT + 8);
    memset(fEstagioAtual,0, sizeof(float) * iNumCT * iNumCT + 8);
    fEstagioAtual[0] = fEstagioAtual[1] = (float)iNumCT;

    for (iEstagio=1; iEstagio<=iNumCT; iEstagio++) {

        iTempoProcessamento
fTempoProcessamento[Index(1,iEstagio,iTempoProcessamentoCol)];
        //iDataMinima = fDataMinima[Index(1,iEstagio,iDataMinimaCol)] ;
        fIP = iTempoProcessamento;
        fEstagioAnt[Index(iEstagio,iEstagio,(int)fEstagioAnt[1])] = fIP;

    }

    //para cada estagio avalia os centros de trabalhos e a melhor
    //alternativa de onde pode vir

```

```

//aloca memoria para para as informacoes da Trajetoria
fTrajetoria = (float*)malloc(sizeof(float) * iNumCT * iNumOP + 8);
memset(fTrajetoria,0, sizeof(float) * iNumCT * iNumOP + 8);
//seta o numero de linhas e colunas do vetor de Trajetoria
fTrajetoria[0] = (float)iNumOP;
fTrajetoria[1] = (float)iNumCT;

//aloca memoria para para as informacoes do Label temporario
fLabel = (float*)malloc(sizeof(float) * iNumCT * 1 + 8);

//Opercao Atual
for (iEstagio=2; iEstagio<=iNumOP; iEstagio++) {

    //limpa as informacoes temporarias do Label
    memset(fLabel,0, sizeof(float) * iNumCT * 1 + 8);
    //seta o numero de linhas e colunas do vetor de Label
    fLabel[0] = (float)iNumCT;
    fLabel[1] = 1;

    //Centro de Trabalho Atual
    for (iCTAtual=1; iCTAtual<=iNumCT; iCTAtual++) {

        //Verifica se Centro de Trabalho permite realizar operacao (CT lido).
        iTempoProcessamento =
fTempoProcessamento[Index(iEstagio,iCTAtual,iTempoProcessamentoCol)];
        if (iTempoProcessamento == 0) continue;
        //Inicializa com um valor alto
        fNorma = 1000000;
        fNormaEuclides = 1000000;
        //Acrescenta o tempo de processamento, ou DataMinima + Processamento
        //iDataMinima = fDataMinima[Index(iEstagio, iCTAtual, (int)iDataMinimaCol)];

        //Varre os centros de trabalho anterior para verificar a melhor alternativa
        for (iCTAnt=1; iCTAnt<=iNumCT; iCTAnt++) {

            //Verifica se Centro de Trabalho anterior valido para Operacao Atual
            if ((fTempoProcessamento[Index(iEstagio - 1, iCTAnt,iTempoProcessamentoCol)] == 0) ||
                (fTrajetoria[Index(iEstagio - 1, iCTAnt,(int)fTrajetoria[1])] == 0) &&
                (iEstagio != 2))
                continue;

            //Faz Label Anterior ser EstagioAtual para o Centro de
            //Trabalho em questao

            CopyVector(fEstagioAnt, iCTAnt, (int)fEstagioAnt[1],
                        fLabel, 1, (int)fLabel);

            iTempoProcessamentoAnt = fLabel[Index(iCTAtual,1,(int)fLabel[1])];
            fIP = iTempoProcessamentoAnt + iTempoProcessamento;
            fLabel[Index(iCTAtual,1,(int)fLabel[1])] = fIP;

            fNormaAux = MaxVector(fLabel);
            fNormaEuclidesAux = NormaVector(fLabel);

            if (fNormaAux <= fNorma) {
                if (fNormaAux == fNorma) {
                    if (fNormaEuclidesAux < fNormaEuclides) {
                        fNorma = fNormaAux;
                        fNormaEuclides = fNormaEuclidesAux;
                        fTrajetoria[Index(iEstagio, iCTAtual,(int)fTrajetoria[1])] =
(float)iCTAnt;
                        CopyVector(fLabel,1,
(int)fLabel[1],fEstagioAtual,iCTAtual,(int)fEstagioAtual[1]);
                    }
                }else{
                    fNorma = fNormaAux;
                    fNormaEuclides = fNormaEuclidesAux;
                    fTrajetoria[Index(iEstagio, iCTAtual,(int)fTrajetoria[1])] =
(float)iCTAnt;
                    CopyVector(fLabel,1,
(int)fLabel[1],fEstagioAtual,iCTAtual,(int)fEstagioAtual[1]);
                }
            }
        }
    }
}

```

```

        }
    }

    } //for iCTAnt
} //for iCTAtual

//limpa as informacoes temporarias do EstagioAnterior
memset(fEstagioAnt, 0, sizeof(float) * iNumCT * iNumCT + 8);
//EstagioAnterior passa a ser EstagioAtual
memcpy(fEstagioAnt,fEstagioAtual,sizeof(float) * iNumCT * iNumCT + 8);

} //for iEstagio

//Seleciona a Melhor Alternativa

//Inicializa a norma com um valor alto
fNorma = 10000;
fNormaEuclides = (float)99999999;

for(iCTAtual=1; iCTAtual<=iNumCT; iCTAtual++) {

    //Verifica se centro de trabalho valido para a Operacao Atual
    iTempoProcessamento = fTempoProcessamento[Index(iNumOP,
iCTAtual,iTempoProcessamentoCol)];
    if (iTempoProcessamento == 0) continue;

    // Verifica se possui mais de um estagio
    if (iNumOP > 1 ) {
        // Verifica se Trajetoria pode ser escolhida
        if (fTrajetoria[Index(iNumOP,iCTAtual,(int)fTrajetoria[1])] == 0)
            continue;
    }

    fNormaAux = MaxVector(fEstagioAtual);
    fNormaEuclidesAux = NormaVector(fEstagioAtual);

    if (fNormaAux < fNorma) {
        fNorma = fNormaAux;
        fSolucao = (float)iCTAtual;
        fNormaEuclides = fNormaEuclidesAux;
    }else{
        if (fNormaAux == fNorma)
            if (fNormaEuclidesAux < fNormaEuclides) {
                fNormaEuclides = fNormaEuclidesAux;
                fNorma = fNormaAux;
                fSolucao = (float)iCTAtual;
            }
    }
} // end iCTAtual

// Reconstrucao da Trajetoria ou alocao de tarefas nos CT

fOp = fSolucao;
fFerramentaOP[Ind1(iNumOP)] = fOp;

iOp = iNumOP;
while (iOp>=2) {
    fOp = fTrajetoria[Index(iOp, (int)fOp,(int)fTrajetoria[1])];
    fFerramentaOP[Ind1(iOp-1)] = fOp;
    iOp--;
}

// Fim da 1a Parte do Sequenciador Matching de Maquinas

} // end metodo

void dbapi004seq2( float *fDataMinima,
                  float *fTempoProcessamento,
                  float *fDisponibilidade,
                  float *fFerramentaOP,
                  float *fCentroTrabalhoOperacao)

```

```

{
int iDataMinimaCol;
int iTempoProcessamentoCol;
int iDisponibilidadeCol;
int iFerramentaCol;
int iCentroTrabalhoOperacaoCol;

int iNumOP;
int iNumCT;
int iCTAtual;
int iCTAnt;
int opAntecessora, iOp;
int iEstagio;

float fNorma = 0;
float fNormaAux = 0;
float fNormaEuclides = 0;
float fNormaEuclidesAux = 0;
float fOp = 0;
float fSolucao = 0;
float fFerramenta = 0;
float iTempoProcessamento = 0;
float iTempoProcessamentoAnt = 0;
float iDataMinima = 0;
float fIP = 0;
float *fEstagioAnt,
      *fEstagioAtual,
      *fCarga,
      *fTrajetoria,
      *fLabel;

Union un;
//FILE *fp;
int iAux,jAux;

//busca o numero de operacoes(iNumOP) e centros de trabalho(iNumCT)
un.f = fDataMinima[0];
iNumOP = un.i;
un.f = fDataMinima[1];
iNumCT = un.i;

//busca o numero de colunas para cada vetor proveniente do progress
iDataMinimaCol = iNumCT;

un.f = fTempoProcessamento[1];
iTempoProcessamentoCol = un.i;

un.f = fDisponibilidade[1];
iDisponibilidadeCol = un.i;

un.f = fFerramentaOP[1];
iFerramentaCol = un.i;

un.f = fCentroTrabalhoOperacao[1];
iCentroTrabalhoOperacaoCol = un.i;

//fp = fopen("c:\\temp\\_dllrell.txt","w");

fEstagioAnt = (float*)malloc(sizeof(float) * iNumCT * iNumCT + 8);
memset(fEstagioAnt,0, sizeof(float) * iNumCT * iNumCT + 8);
fEstagioAnt[0] = fEstagioAnt[1] = (float)iNumCT;
fEstagioAtual = (float*)malloc(sizeof(float) * iNumCT * iNumCT + 8);
memset(fEstagioAtual,0, sizeof(float) * iNumCT * iNumCT + 8);
fEstagioAtual[0] = fEstagioAtual[1] = (float)iNumCT;

for (iEstagio=1; iEstagio<=iNumCT; iEstagio++) {

    iTempoProcessamento
fTempoProcessamento[Index(1,iEstagio,iTempoProcessamentoCol)];
    iDataMinima = fDataMinima[Index(1,iEstagio,iDataMinimaCol)] ;
    fIP = iTempoProcessamento;
    fEstagioAnt[Index(iEstagio,iEstagio,(int)fEstagioAnt[1])] = fIP;

}

```

```

//para cada estagio avalia os centros de trabalhos e a melhor
//alternativa de onde pode vir

//aloca memoria para para as informacoes de Carga
fCarga = (float*)malloc(sizeof(float) * iNumCT * 1 + 8);
memset(fCarga,0, sizeof(float) * iNumCT * 1 + 8);
//seta o numero de linhas e colunas do vetor de Carga
fCarga[0] = (float)iNumCT;
fCarga[1] = 1;

//aloca memoria para para as informacoes da Trajetoria
fTrajetoria = (float*)malloc(sizeof(float) * iNumCT * iNumOP + 8);
memset(fTrajetoria,0, sizeof(float) * iNumCT * iNumOP + 8);
//seta o numero de linhas e colunas do vetor de Trajetoria
fTrajetoria[0] = (float)iNumOP;
fTrajetoria[1] = (float)iNumCT;

//aloca memoria para para as informacoes do Label temporario
fLabel = (float*)malloc(sizeof(float) * iNumCT * 1 + 8);

//Opercao Atual
for (iEstagio=2; iEstagio<=iNumOP; iEstagio++) {

    //limpa as informacoes temporarias do Label
    memset(fLabel,0, sizeof(float) * iNumCT * 1 + 8);
    //seta o numero de linhas e colunas do vetor de Label
    fLabel[0] = (float)iNumCT;
    fLabel[1] = 1;

    //verifica se existe ferramenta antecessora para a operacao atual
    fFerramenta = fFerramentaOP[Index(iEstagio)];
    if (fFerramenta)
        opAntecessora = FerramentaAntecessora(iEstagio, fFerramenta, fFerramentaOP);
    else
        opAntecessora = 0;

    //Centro de Trabalho Atual
    for (iCTAtual=1; iCTAtual<=iNumCT; iCTAtual++) {

        //Verifica se Centro de Trabalho permite realizar operacao (CT lido).
        iTempoProcessamento
fTempoProcessamento[Index(iEstagio,iCTAtual,iTempoProcessamentoCol)];
        if (iTempoProcessamento == 0) continue;
        //Inicializa com um valor alto
        fNorma = 1000000;
        fNormaEuclides = 1000000;
        //Acrescenta o tempo de processamento, ou DataMinima + Processamento
        iDataMinima = fDataMinima[Index(iEstagio, iCTAtual, (int)iDataMinimaCol)];

        //Varre os centros de trabalho anterior para verificar a melhor alternativa
        for (iCTAnt=1; iCTAnt<=iNumCT; iCTAnt++) {

            //Verifica se Centro de Trabalho anterior valido para Operacao Atual
            if ((fTempoProcessamento[Index(iEstagio - 1, iCTAnt,iTempoProcessamentoCol)]
== 0) ||
                (fTrajetoria[Index(iEstagio - 1, iCTAnt,(int)fTrajetoria[1])] == 0) &&
                (iEstagio != 2))
                continue;

            //Se para a alternativa avaliada, a operacao antecessora que usa o
            //ferramental nao eh do mesmo centro de trabalho, desconsiderar.
            if ((fFerramenta) && (opAntecessora != 0) &&
                !( ProcessaTrajetoria(iCTAtual, iCTAnt, iEstagio, opAntecessora,
fTrajetoria)) )
                continue;

            //Faz Label Anterior ser EstagioAtual para o Centro de
            //Trabalho em questao

            CopyVector(fEstagioAnt, iCTAnt, (int)fEstagioAnt[1],
                    fLabel, 1, (int)fLabel);

            iTempoProcessamentoAnt = fLabel[Index(iCTAtual,1,(int)fLabel[1])];
            fIP = iTempoProcessamentoAnt + iTempoProcessamento;
            fLabel[Index(iCTAtual,1,(int)fLabel[1])] = fIP;

```



```

        DivideVector(fLabel,1,(int)fLabel[1],
                    fDisponibilidade,1, iDisponibilidadeCol,
                    fCarga,1, (int)fCarga[1]);

        fNormaAux = MaxVector(fCarga);

        fNormaEuclidesAux = NormaVector(fCarga);

        if (fNormaAux <= fNorma) {
            if (fNormaAux == fNorma) {
                if (fNormaEuclidesAux < fNormaEuclides) {
                    fNorma = fNormaAux;
                    fNormaEuclides = fNormaEuclidesAux;
                    fTrajetoria[Index(iEstagio, iCTAtual,(int)fTrajetoria[1])] =
(float)iCTAnt;
                    CopyVector(fLabel,1,
(int)fLabel[1],fEstagioAtual,iCTAtual,(int)fEstagioAtual[1]);
                }
            }else{
                fNorma = fNormaAux;
                fNormaEuclides = fNormaEuclidesAux;
                fTrajetoria[Index(iEstagio, iCTAtual,(int)fTrajetoria[1])] =
(float)iCTAnt;
                CopyVector(fLabel,1,
(int)fLabel[1],fEstagioAtual,iCTAtual,(int)fEstagioAtual[1]);
            }
        }
    } //for iCTAnt
} //for iCTAtual

/*fprintf(fp,"\n\n Estagio %i \n",iEstagio);
fprintf(fp,"Estagio Anterior      EstagioAtual\n\n");
for( iAux = 1; iAux<=iNumCT; iAux++ )
    for( jAux = 1; jAux <= iNumCT; jAux++ )
        fprintf(fp,"[%2i] [%2i]      %10.3f      [%2i] [%2i]      %10.3f      \n", iAux,
jAux, fEstagioAnt [Index(iAux,jAux,(int)fEstagioAnt[1])],
        iAux, jAux, fEstagioAtual [Index(iAux,jAux,(int)fEstagioAtual[1])])
);
*/
//limpa as informacoes temporarias do EstagioAnterior
memset(fEstagioAnt,0, sizeof(float) * iNumCT * iNumCT + 8);
//EstagioAnterior passa a ser EstagioAtual
memcpy(fEstagioAnt,fEstagioAtual,sizeof(float) * iNumCT * iNumCT + 8);

} //for iEstagio

//Seleciona a Melhor Alternativa
//Inicializa a norma com um valor alto
fNorma = 100000;
fNormaEuclides = (float)99999999;

for(iCTAtual=1; iCTAtual<=iNumCT; iCTAtual++) {

    //Verifica se centro de trabalho valido para a Operacao Atual
    iTempoProcessamento = fTempoProcessamento[Index(iNumOP,
iCTAtual,iTempoProcessamentoCol)];
    if (iTempoProcessamento == 0) continue;

    // Verifica se possui mais de um estagio
    if (iNumOP > 1 ) {
        // Verifica se Trajetoria pode ser escolhida
        if (fTrajetoria[Index(iNumOP,iCTAtual,(int)fTrajetoria[1])] == 0.0)
            continue;
    }

    //Metrica de Avaliacao
    DivideVector(fEstagioAtual, iCTAtual,(int)fEstagioAtual[1],
                fDisponibilidade, 1, iDisponibilidadeCol,
                fCarga, 1, (int)fCarga[1]);

    fNormaAux = MaxVector(fCarga);

```

```

        fNormaEuclidesAux = NormaVector(fCarga);

        if (fNormaAux < fNorma) {
            fNorma = fNormaAux;
            fSolucao = (float)iCTAtual;
            fNormaEuclides = fNormaEuclidesAux;
        } else {
            if (fNormaAux == fNorma)
                if (fNormaEuclidesAux < fNormaEuclides) {
                    fNormaEuclides = fNormaEuclidesAux;
                    fNorma = fNormaAux;
                    fSolucao = (float)iCTAtual;
                }
        }
    } // end iCTAtual

    // Reconstrucao da Trajetória ou alocação de tarefas nos CT
    /*fprintf(fp, "\nTRAJETORIA\n");
    for( iAux = 1; iAux<=iNumOP; iAux++ ) {
        fprintf(fp, "\n\n");
        for( jAux = 1; jAux <= iNumCT; jAux++ )
            fprintf(fp, "[%2i] [%2i] %3f \n", iAux, jAux, fTrajetoria
[Index(iAux, jAux, (int)fTrajetoria[1])]);
    }*/

    fOp = fSolucao;
    fCentroTrabalhoOperacao[Indl(iNumOP)] = fOp;

    iOp = iNumOP;
    while (iOp>=2) {
        fOp = fTrajetoria[Index(iOp, (int)fOp, (int)fTrajetoria[1])];
        fCentroTrabalhoOperacao[Indl(iOp-1)] = fOp;
        iOp--;
    }

    /*for(iOp=1; iOp<=iNumOP; iOp++)
        fprintf(fp, "Operação: %i Centro: %4.4f
\n", iOp, fCentroTrabalhoOperacao[Indl(iOp)]); */

    //fclose(fp);

    free(fLabel);
    free(fEstagioAnt);
    free(fEstagioAtual);
    free(fCarga);
    free(fTrajetoria);

    // Fim da 1a Parte do Sequenciador Matching de Maquinas
} // end metodo

/*****
METODO: dbapi004seq1
DESCRICAO: Segunda Parte do Sequenciador Alocação de Centros de Trabalho
Prioridade
PARAMETROS: void
RETORNO: void
OBSERVACAO:
*****/
void dbapi004seq1(
    float *DataMinima,
    float *DataEntrega,
    float *TempoProcessamento,
    float *Disponibilidade,
    float *Prioridade,
    float *Ferramenta_OP,
    float *CTdaOP,
    float AtrasoSignificativo) {

    int iOper, Centro_Trabalho, Oper_Antecessora;

```

```

int iNumCT, iNumOP, iCentro;
double fPrioridade, AtrasoRelativo, ATC;
float *Evolucao;
float fBase;
float Gatilho;
int iNC;
//FILE *arq;
Union u;
//char s[256];

//arq = fopen("c:\\temp\\_seq01.txt", "w");

u.f = DataMinima[0];
iNumOP = u.i;

u.f = DataMinima[1];
iNumCT = u.i;

iNC = iNumCT;

//fBase = (float) pow(10, 2/AtrasoSignificativo);

Evolucao = (float *) malloc(sizeof(float)*(iNumCT + 1));
memset(Evolucao, 0, sizeof(float)*(iNumCT + 1));
if( Evolucao == NULL ) {
    printf( "Impossível Alocar para Evolucao" );
    return;
}

//sprintf(s, "Atraso: %10.6f \n" , AtrasoSignificativo);
//fprintf(arq, s);

for( iOper = 1; iOper <= iNumOP; iOper++ ) {

    //Se operação usa Ferramenta então
    //encontra operação antecessora que usa a mesma ferramenta.
    if( Ferramenta_OP[ Ind1(iOper) ] )
        Oper_Antecessora = (int) FerramentaAntecessora(iOper, Ferramenta_OP[
Ind1(iOper) ],
                                Ferramenta_OP );
    else
        Oper_Antecessora = 0;

    //fprintf(arq, "Operação: %i \n Antecessora: %i \n", iOper, Oper_Antecessora);

    if( Oper_Antecessora ) {
        //Atribui o mesmo centro de trabalho da operação antecessora que
        //utiliza a mesma ferramenta
        Centro_Trabalho = (int) CTdaOP[ Ind1(Oper_Antecessora) ] ;
        CTdaOP[ Ind1(iOper) ] = (float) Centro_Trabalho;
        //Atualiza a data de início da próxima operação no centro de trabalho que
        //recebeu a operação
        Evolucao[Centro_Trabalho] += TempoProcessamento[Ind2(iOper, Centro_Trabalho)];

        if( DataMinima[ Ind2(iOper, Centro_Trabalho) ] > Evolucao[Centro_Trabalho] )
            Evolucao[Centro_Trabalho] = DataMinima[ Ind2(iOper, Centro_Trabalho) ];

        continue;
    }

    //Para cada Centro de Trabalho, calcula prioridade aparente para operação
    //guarda o máximo para alocação
    fPrioridade = -1;
    Centro_Trabalho = 0;
    //fprintf(arq, "%3s %12s %12s %12s %3s\n",
    //        "Cen", "AR", "ATC", "fPrio", "CT");

    for( iCentro=1; iCentro<=iNumCT; iCentro++ ) {

        if( !TempoProcessamento[ Ind2(iOper, iCentro) ] ) continue;

        AtrasoRelativo = (Evolucao[ iCentro ] + TempoProcessamento [
Ind2(iOper, iCentro) ] -
                        DataEntrega[ Ind2(iOper, iCentro) ] ) / Disponibilidade[
Ind1(iCentro) ];

```

```

        Gatilho          = - AtrasoSignificativo / Disponibilidade[ Ind1(iCentro) ];
        Gatilho          += 1.0 / log10(2);

        ATC = Prioridade[ Ind2(iOper,iCentro) ] * pow( (double) 2.0, -
(MAX(0,AtrasoRelativo + Gatilho)) );
        /*sprintf(s, "Oper: %3i Centro: %3i Prioridade: %3.0f Dispon: %6.2f Tempo:
%6.2f Atraso: %10.6f ATC: %10.6f\n",
                iOper, iCentro, Prioridade[ Ind2(iOper,iCentro) ], Disponibilidade[
Ind1(iCentro) ],
                TempoProcessamento [ Ind2(iOper,iCentro) ], AtrasoRelativo, ATC);
        fprintf(arq,s);*/

        if(ATC > fPrioridade ) {
            fPrioridade = ATC;
            Centro_Trabalho = iCentro;
        }
        //fprintf(arq,"%3i %12.9f %12.9f %12.9f %12.9f %12.9f %3i\n",
        // iCentro, AtrasoRelativo, fBase, ATC, pow( (double) fBase, (double) -
(MAX(0,AtrasoRelativo)) ), fPrioridade, Centro_Trabalho);
    }
    CTdaOP[ Ind1(iOper) ] = (float) Centro_Trabalho;
    //Atualiza a data de início da próxima operação no centro de trabalho que
    //recebeu a operação
    Evolucao[Centro_Trabalho] += TempoProcessamento[Ind2(iOper,Centro_Trabalho)];
    if( DataMinima[ Ind2(iOper,Centro_Trabalho)] > Evolucao[Centro_Trabalho] )
        Evolucao[Centro_Trabalho] = DataMinima[ Ind2(iOper,Centro_Trabalho)];
}

free(Evolucao);
//fclose(arq);
}

/*****
METODO: dbapi004seq3
DESCRICAO: Segunda Parte do Sequenciador
PARAMETROS: void
RETORNO: void
OBSERVACAO:
*****/

void dbapi004seq3( float *DataMinima,
                  float *DataEntrega,
                  float *TempoProcessamento,
                  float *Prioridade_OP,
                  float *CTdaOP,
                  float *Sequencia,
                  float k)
{
    int      iOper,CT;
    int      iNumCT, iNumOP,iCentro;
    double   fPrioridade, Atraso, ATC;
    float *Controle;
    float fTM, fTP, fDM, Expo;
    int      iNC;

    /*****
    O vetor Sequenciada possui a informação se a operação já foi Sequenciada
    O vetor Sequencia possui qual a posição da ordem no Centro de Trabalho
    *****/
    int      *Sequenciada, iOrdem, Num_OP_Sequenciadas;
    //FILE    *arq;
    Union u;

    /*****
    Carrega o Número de Operações e Centros de Trabalho
    *****/
    u.f      = DataMinima[0];
    iNumOP   = u.i;
    u.f      = DataMinima[1];
    iNumCT   = u.i;
    //iNumOP = (int) DataMinima[0];
    //iNumCT = (int) DataMinima[1];

```

```

iNC = iNumCT;
/*****/

/*****/

Prepara os Vetores
*****/

Controle = (float *) malloc( sizeof(float) * (iNumCT+1) );
Sequenciada = (int *) malloc( sizeof(int) * (iNumOP+1) );

memset(Controle, 0, sizeof(float)*6*(iNumCT+1));
memset(Sequenciada, 0, sizeof(float)*(iNumOP+1));

/*****/
//arq = fopen("c:\\operacoes.txt","a");

/*****/
Conta o número de operações e faz a somatória dos tempos de processamento
*****/

for(iOper=1;iOper<=iNumOP;iOper++) {
    CT = (int) CTdaOP[ Ind1(iOper) ] ;
    Controle[ Ind(CT, NOPC) ] ++;
    Controle[ Ind(CT, TM ) ] += TempoProcessamento[ Ind2(iOper,CT) ] ;
}

iOrdem = 1;
Num_OP_Sequenciadas = iNumOP;

while( Num_OP_Sequenciadas > 0 ) {
    /*****/
    Prepara o Centro para identificar a próxima operação
    *****/

    for(iCentro=1;iCentro<=iNumCT;iCentro++) {
        Controle[ Ind(iCentro, MATC) ] = 0;
        Controle[ Ind(iCentro, OP ) ] = 0;
    }

    /*****/
    Calcula o ATC e Encontra o Maior, identificando a operação crítica
    *****/

    for(iOper=1;iOper<=iNumOP;iOper++) {
        if(Sequenciada[ iOper ]) continue;
        CT = (int) CTdaOP[ Ind1(iOper) ] ;
        Atraso = DataEntrega[ Ind2(iOper, CT) ] -
            Controle[ Ind(CT, IP) ] - TempoProcessamento [ Ind2(iOper,CT) ];
        //Atraso = MAX(0,Atraso);
        fTM = Controle[ Ind(CT,TM) ] /
            ( Controle[ Ind(CT,NOPC) ] - Controle[ Ind(CT,NOPS) ] );
        fPrioridade = Prioridade_OP[ Ind1(iOper) ];
        fTP = TempoProcessamento [ Ind2(iOper,CT) ];
        fDM = DataMinima[ Ind2(iOper,CT) ] - Controle[ Ind(CT, IP) ];
        fDM = MAX(0,fDM);
        Expo = Atraso > 0 ? Atraso / fTM : Atraso / 24 ;
        ATC = fPrioridade/fTP * pow( 2, - Expo)*pow(10,- fDM);
        /*fprintf(arq,"%3i %3i %-8.2f %-8.2f %-8.2f %-8.2f %-8.2f %-8.2f %-8.2f %-8.2f %-8.2f\n",
            iOper,
            CT,
            DataEntrega[ Ind2(iOper, CT) ],
            Controle[ Ind(CT, IP) ],
            TempoProcessamento [ Ind2(iOper,CT) ],
            Atraso,
            fTM,
            fPrioridade,
            fTP,
            fDM,
            ATC );*/

    /*****/
    Verifica se Operação é a mais Crítica
    *****/

    if( ATC >= Controle[ Ind(CT, MATC) ] ) {
        Controle[ Ind(CT, MATC) ] = (float)ATC;

```

```

        Controle[ Ind(CT, OP ) ] = (float)iOper;
    }
}

/*****
Sequencia operação crítica escolhida e
atualiza instante em que o centro ficará livre
*****/

for(iCentro=1;iCentro<=iNumCT;iCentro++) {
    iOper = (int)Controle[ Ind(iCentro, OP) ];

    if(!iOper) continue;

    Sequenciada[ iOper ] = 1;

    if( DataMinima[ Ind2(iOper,iCentro) ] > Controle[ Ind(iCentro, IP) ] )
        Controle[ Ind(iCentro, IP) ] = DataMinima[ Ind2(iOper,iCentro) ] +
            TempoProcessamento[ Ind2(iOper,iCentro) ];
    else
        Controle[ Ind(iCentro, IP) ] += TempoProcessamento[ Ind2(iOper,iCentro) ];

    Controle[ Ind(iCentro, NOPS) ] ++;

    Sequencia[ Ind1(iOper) ] = (float)iOrdem;
    Num_OP_Sequenciadas --;

    //Atualiza Somatória dos tempos das operações faltantes
    Controle[ Ind(iCentro, TM) ] -= TempoProcessamento[ Ind2(iOper,iCentro) ] ;

}

iOrdem ++;
}
//fclose(arq);
//free(Controle);
free(Sequenciada);
}

```